Adaptive Computer Assisted Assessment of free-text students' answers: an approach to automatically generate students' conceptual models

Diana Rosario Pérez Marín
under the supervision of Enrique Alfonseca
and Pilar Rodríguez

May, 2007

Contents

Abstract **xiii**

Resumen **xv**

Acknowledgements **xvii**

Acronyms **xix**

Publications **xxi**

Some explanatory notes **1**

1 Introduction **3**
1.1 Goals 4
1.2 A sample scenario 4
1.3 Thesis overview 6
1.3.1 General idea 6
1.3.2 Context 7
1.3.3 Organization 7

I State-of-the-art review **9**

2 Some cognitive and pedagogic theories **13**
2.1 Definition of concepts and their categories 13
2.2 From concepts to words 15
2.3 Relationships among concepts 16
2.4 Mental models 17
2.5 Ausubel's Meaningful Learning Theory 18
2.6 Novak's concept maps as a tool of Ausubel's theory 19
2.7 Ausubel and Novak ideas in practise 22

3 Students' conceptual modeling **23**
3.1 Points of view about students' models 24
3.2 Representation forms of students' conceptual models 25

3.3 Methods to build and update student models . . . 26
3.4 Some systems underpinned by conceptual models . . . 29
3.5 Comparison and conclusions . . . 34

4 Computer Assisted Assessment of free-text answers 37
4.1 Techniques . . . 39
4.2 Evaluation procedures . . . 43
4.2.1 Requisites . . . 44
4.2.2 Metrics . . . 45
4.3 Existing free-text CAA systems . . . 46
4.3.1 AEA . . . 46
4.3.2 Apex Assessor . . . 47
4.3.3 ATM . . . 48
4.3.4 Automark . . . 48
4.3.5 Auto-marking . . . 49
4.3.6 BETSY . . . 50
4.3.7 C-rater and E-rater . . . 50
4.3.8 CarmelTC . . . 51
4.3.9 EGAL . . . 52
4.3.10 IEA . . . 52
4.3.11 IEMS . . . 53
4.3.12 IntelliMetric . . . 54
4.3.13 Jess . . . 54
4.3.14 Larkey's system . . . 55
4.3.15 MarkIT . . . 56
4.3.16 MRW . . . 56
4.3.17 PEG . . . 57
4.3.18 PS-ME . . . 57
4.3.19 RMT . . . 58
4.3.20 SAGrader . . . 59
4.3.21 SEAR . . . 59
4.4 Comparison and conclusions . . . 60

II Proposal for the generation of students' conceptual models underpinned by free-text Adaptive Computer Assisted Assessment (ACAA) 63

5 Automatic and adaptive free-text assessment for conceptual modeling 67
5.1 Scope . . . 68
5.2 Domain Model . . . 68
5.3 Student Model . . . 70
5.3.1 Static component of the model . . . 71

5.3.2 Dynamic component of the model . . . 72
5.4 Generation of the conceptual and domain model . . . 75
5.4.1 The concepts are found . . . 75
5.4.2 The type 1 and 2 relationships between the concepts are found . . . 77
5.4.3 The free-text ACAA system is used by the student . . . 77
5.4.4 The confidence-value of each student's concepts is calculated . . . 79
5.4.5 The type 3 relationships between concepts are found . . . 82
5.4.6 The conceptual model is updated by instructors, the free-text ACAA system and/or students . . . 83
5.5 Class conceptual model . . . 86

6 An example of free-text ACAA system: Willow 87
6.1 Non-adaptive version of Willow: Atenea . . . 87
6.2 Willow's main features . . . 89
6.3 Willed . . . 94
6.3.1 Creation of a new area-of-knowledge . . . 94
6.3.2 Modification of an existing area-of-knowledge . . . 96
6.4 High-level architecture . . . 98
6.5 Low-level architecture . . . 101
6.5.1 Processing module . . . 101
6.5.2 Comparison module . . . 102
6.5.3 Feedback module . . . 109
6.6 Optimum use of Willow . . . 110

7 An example of conceptual viewer: COMOV 113
7.1 Concept map . . . 114
7.2 Conceptual diagram . . . 118
7.3 Table . . . 120
7.4 Bar chart . . . 121
7.5 Textual summary . . . 123
7.6 Recap of the main points of the five representation formats . . . 125

III Evaluation and conclusions 127

8 Experiments and evaluation 131
8.1 First experiment . . . 132
8.2 Second experiment . . . 138
8.3 Third experiment . . . 140
8.4 Other experiments . . . 142
8.4.1 ERB . . . 144
8.4.2 Comparison of ERB with baselines . . . 148

8.4.3 NLP+ERB . . . 150
8.4.4 ERB+LSA . . . 152
8.4.5 ERB+NLP+LSA . . . 154
8.4.6 ERB+Genetic Algorithms . . . 155
8.4.7 RARE+Willow . . . 156
8.4.8 Automatic Term Identification . . . 157

9 Conclusions and future work **159**
9.1 Fulfilled goals for teachers . . . 160
9.2 Fulfilled goals for students . . . 161
9.3 Fulfilled subgoals . . . 162
9.4 Comparison to related systems . . . 164
9.5 Extending the procedure to other language and/or domain . . . 166
9.6 Future work . . . 168

Appendix A: Engineering work **171**

Appendix B: Data of the experiments **179**

Appendix C: Introduction (in Spanish) **183**

Appendix D: Conclusions (in Spanish) **191**

Appendix E: Examples (in Spanish) **203**

References **205**

Web References **225**

List of Figures

1.1 Picture of the approach proposed in this work. 6
1.2 Fields to which this work is related and how they are related among themselves. Notice that the most shadowed fields are the ones that have been more addressed. 7

2.1 Overview of the Ausubel's meaningful learning theory. 18
2.2 An example of Novak's concept map (source: Novak et al., 1984). 20

3.1 An example of a tree structure on the left and of conceptual graph on the right. . 25
3.2 Snapshot of Dynmap. 31
3.3 Snapshot of an E-TESTER output. 31
3.4 A snapshot of the CourseVis system. 33

4.1 Time line of research in CAA for free-text answers. 38
4.2 Example of a scheme used in Automark to score the answer to the question like "*What movement relates the Earth and the Sun?*" (Mitchell et al., 2003). 43
4.3 Architecture of the ATM system (Callear et al., 2001). 48
4.4 A snapshot of the MRW system. 56

5.1 Representation of the proposed storage for the domain model as a simplified entity-relation model. Horizontal links should be read from left to right and vertical links from top to bottom. 69
5.2 An example of adaptation paths from the features gathered in Table 5.1. The whole branching is not shown, instead, asterisks are used to indicate that the same branching should be repeated where they appear (the number of asterisks indicates which branch should be copied). 72
5.3 Example of the hierarchical structure of the conceptual model. 74
5.4 Example of training file for the algorithm to automatically extract the terms from the references. 76

6.1 A question delivered by the system according to the settings provided. 88
6.2 A snapshot of Willow's configuration session stage. 89
6.3 A snapshot of the login page of Atenea and of page delivering a question. 90
6.4 A snapshot of Willow's feedback page. 92

6.5 On the left, a snapshot of Willoc and on the right, the Willow's personalization possibilities. . . . 93
6.6 On the top, a snapshot of the Willed's page with the initial menu (to modify an already area-of-knowledge collection of questions or to create a new one) and at the bottom, the page to create a new area-of-knowledge. . . . 95
6.7 At the top, a snapshot of the Willed's page to modify an already existing collection of questions and at the bottom, the page to create a new one question. . . . 97
6.8 At the top, a snapshot of the Willed's page to modify an already existing question and at the bottom, the page to modify its references. . . . 98
6.9 Example of the generation of new references from the original text "*Unix is an operating system. It is multi-user. It is easy to use*". . . . 99
6.10 A simplified diagram of Willow's high-level architecture. . . . 100
6.11 Modification of a student's answer depending on the configuration of Willow. The synset identifiers in the last four cases are taken from WordNet 1.7. . . . 103
6.12 Procedure for calculating the Modified Brevity Penalty factor. . . . 105

7.1 A snapshot of the COMOV's login page. . . . 114
7.2 An example of a good student's conceptual model represented as a concept map using CLOVER. . . . 115
7.3 An example of a bad student's conceptual model represented as a concept map using CLOVER. . . . 116
7.4 An example of concept map of a student using IOV. . . . 117
7.5 An example of concept map of a class that has started a course. . . . 118
7.6 An example of the concept map of the same class but some months later. . . . 119
7.7 An example of conceptual model represented as a conceptual diagram. . . . 120
7.8 An example of table ordered from higher to lower confidence-value basic-concepts of a student's conceptual model. . . . 122
7.9 An example of bar chart ordered from lower to higher confidence-value basic-concepts of a student's conceptual model. . . . 123
7.10 An example of textual summary of a student's conceptual model. . . . 124

8.1 Results of the survey carried out about the usability and usefulness of Willed. . . . 133
8.2 Number of questions answered week by week from October 2006 to January 2007 by the students of the third experiment. . . . 141
8.3 Regression lines between the teachers' scores and the automatic marks for sets 2 and 5. . . . 145
8.4 Regression lines between the teachers' scores and the automatic marks for sets 6 and 7. . . . 146
8.5 Histogram for definition datasets (2 and 3). . . . 147
8.6 Histograms for datasets 5 and 7. . . . 148

1 Simplified Entity-Relationship model of the Willow's database. . . . 171

2 Representación gráfica del procedimiento descrito en este trabajo. 186
3 Campos relacionados con este trabajo y cómo se enmarca este trabajo entre ellos 187

List of Tables

3.1 Comparison table between three different knowledge representation forms. 26
3.2 Comparison table between different techniques to build students' models. In the goal column: P. indicates prediction, R. recommendation, C. classification and D. detection and in the granularity column: I means for modeling individuals and G for modeling groups. 29
3.3 First comparison table between ten different systems underpinned by conceptual models. 34
3.4 Second comparison table between ten different systems underpinned by conceptual models. 35

4.1 Review of the six Bloom's competence levels, the main skill that they demonstrate, two examples of question cues and a relevant assessment method for each of them (source: Bishop, 2002). 38
4.2 Lecturers' expectations about what a CAA of free text answers software system should provide them in order to be useful (Darus et al., 2001). 44
4.3 Students' expectations about what a CAA of free text answers software system should provide them in order to be useful (Darus et al., 2001). 44
4.4 Domains to which the current existing CAA of free text answers systems have been applied and their availability. 60
4.5 Overview of the techniques, evaluation and language of the reviewed free-text CAA systems. Possible metrics are: Corr, correlation; Agr, Agreement; EAgr, Exact Agreement; CAcc, Classification accuracy; f-S, f-Score; and, – for not available. When the authors have presented several values for the results, the mean value has been taken. 61

5.1 Example of values (two have been shown per feature but it could be a greater number) for a set of features proposed for a free-text ACAA system so that it can offer static adaptation. 71
5.2 Example of ten terms automatically identified for the example Operating System course and their frequency in all the references provided by the instructors. . . . 76

6.1 Main differences between Atenea and Willow. 88
6.2 Domain in which Willow has been applied and its availability. 89

6.3 Techniques used in Willow, languages that it can process and its result measured as the Pearson correlation between the teachers' and Willow's scores for the corpus used as explained in Chapter 8. 90
6.4 Example of Domain Matrix. 107

7.1 COMOV values for the second comparison table presented in Chapter 3. 114
7.2 Comparison of several representation formats of the conceptual model. The Figures indicated per each of them represents the same conceptual model of a student enrolled in an Operating System course. 126

8.1 Degree of familiarity of the authors with authoring tools. 132
8.2 Average results for the first experiment in which a group of students (A) used Atenea and, another group of students (B) used Willow in one of their classes. . 134
8.3 Results of the satisfaction survey carried out for six teachers of the Universidad Autónoma de Madrid and their average values in the mean column. Notice that the concept map representation is marked as C, the table as T, the bar chart as B and the text summary as S. 135
8.4 Results of the analysis of the first sixteen generated students' conceptual models. 137
8.5 Results of the remaining 17-31 generated students' conceptual models. 137
8.6 Summary of the logs gathered in the second experiment in which students could use Atenea (non-adaptive sessions) or Willow (adaptive sessions) without any restriction during a week. 139
8.7 Summary of the logs gathered in the third experiment in which students could use Atenea (non-adaptive sessions) or Willow (adaptive sessions) without any restriction during the whole course. 140
8.8 Answer sets used in the evaluation. Rows indicate: set number, number of candidate texts (No. cands.), mean length of the candidate texts in words (Mean cands.), number of reference texts (No. refs.), mean length of the reference texts (Mean refs.), question type (Def., definitions and descriptions; A/D, advantages and disadvantages; Y/N, Yes-No and justification), range of scores as provided by the instructors and source language in which they were written (En., English and Sp., Spanish). 142
8.9 Scores of BLEU for a varying number of reference texts (using the source datasets in Spanish except the first one, which is in English). 144
8.10 Comparison of ERB results when using the MBP with trigrams, bigrams and unigrams or only unigrams against the original Papineni et al. BP factor. The source datasets are all in Spanish except the first one, which is in English. 145
8.11 Mean quadratic error for the several regression lines. 147
8.12 Comparison of ERB with two other keyword-based methods. Because of the five-fold evaluation, datasets with very few answers could not be evaluated with VSM. The Spanish source sets are used. 149

8.13 Comparison of ERB with LSA. The English translated sets are used. 150
8.14 Correlations achieved by different combinations of NLP techniques for the original and translated datasets (the comparison module using ERB is always used too). The *Sp* suffix indicates that the set is in Spanish and *En* that it is in English. . . 151
8.15 Number of different words in the Spanish and English datasets. Row DifV shows the percentage of vocabulary reduction due to the translation. 152
8.16 Correlation between the percentage of reduction of the variability of the vocabulary and the system's performance for different configurations. 152
8.17 Evaluation of the combined systems fixing $\alpha = 0.5$. Cells report the correlations. Please, notice that these results are for the English translated texts. 153
8.18 Evaluation of the combined systems by optimizing the parameter α. Cells report the mean correlations and the values of α at the bottom. The datasets used are the English translated versions. 153
8.19 Results of the combination of ERB+NLP+LSA with $\alpha = 0.5$ and using the CS corpus to train LSA. The English translations of the datasets are used. 154
8.20 ERB+NLP+LSA with α = 0.174, 0.346, 0.323, 0.151 and 0.298, respectively. The CS corpus to train LSA is used. The English translated datasets are used. . 154
8.21 The correlation values obtained while using the original ERB and ERB with the best choice of reference texts that was found by the genetic algorithm. Column 2 shows the number of references that were selected in each case. The datasets used are the original datasets in Spanish. 155
8.22 Results achieved using RARE+Willow to reduce the lexical variability. The datasets used are the English translations. 156
8.23 Results achieved using RARE+Willow to generate new references. Column NGR indicates the number of automatically generated references. The datasets used are the English translations. 156
8.24 Results achieved by Willow using several NLP modules and the method of manually generating new references (MGR). The datasets used are the English translations. 157
8.25 Results of using C4.5 (70% training and 30% test) to automatically identify terms. 157

9.1 List of features. 164
9.2 Systems from the ones reviewed in Chapters 3 and 4 that fulfil the features identified in Table 9.1. 164
9.3 Natural Language Processing techniques that can be used by the procedure (the techniques that are essential for the procedure are marked with an asterisk) and the languages in which they are currently available in Willow. 166
4 Lista de características. 197
5 Sistemas de los revisados en los capítulos 3 y 4 que cumplen las características identificadas en la Tabla 4. 198

6 Técnicas de Procesamiento de Lenguaje Natural (PLN) que se pueden usar para el procedimiento de generación de los modelos conceptuales (las técnicas que son obligatorias están marcadas con un asterisco) y, los idiomas en los que está actualmente disponibles en Willow. 199

Abstract

Teachers aim to transmit their knowledge to students so that they acquire certain accepted and shared concepts. However, what students actually understand is, in many cases, something completely different. Students build their own conceptual models as a network of interrelated concepts depending on their particular background and emotions. In fact, according to the Ausubel's Learning Theory, students will only be able to learn new concepts provided that they have the previous necessary concepts to which the new ones have to be linked to.

Intelligent Tutoring Systems (ITSs) are educational software containing some kind of intelligent component to imitate how a human teacher would behave when teaching. Adaptive Educational Hypermedia systems (AEHSs) are inspired in ITSs to adapt the content and navigation in the course to each student's model. All the same, the assessment of these systems is usually focused on the so-called objective testing (Multiple-Choice Questions, fill-in-the-blank items, etc.), which is commonly agreed that fails to identify many students' deep underlying misconceptions. Hence, automatic assessment of free-text answers is a field that has attracted much attention in the last decades. About twenty different systems are being used both in academic and commercial environments underpinned by several Natural Language Processing (NLP) techniques. Nevertheless, none of these systems keeps any kind of student model.

It has been highlighted the importance of keeping a student model that serves to identify the lack of previous knowledge and to let the system decide which questions are the most suitable. Therefore, in this work, ideas from the AEH, NLP and automatic assessment of free-text answers fields are combined to build a bridge between what teachers try to transmit and what students actually understand. It is achieved by a new procedure able to automatically generate the students' conceptual models from their answers to a free-text Adaptive Computer Assisted Assessment (ACAA), which is the evolution of free-text Computer Assisted Assessment systems but incorporating the use of a student model to adapt the assessment. Using this approach, it is possible to generate each particular student's conceptual model and the whole class conceptual model.

That way, teachers can visualize, at any time, the degree of assimilation of the concepts exposed in the lessons and, to discern their students' faulty or incomplete knowledge in order to organize more efficiently the agenda of their courses. Furthermore, by using the free-text ACAA system, students can get instant feedback from their answers (score, processed answer and correct answers provided by the teachers) and, by looking at their conceptual models, they can organize their study, i.e. review which concepts have already assimilated and, which ones are still missing or are wrongly connected to other concepts.

The procedure has been implemented in the *Will tools* that consist of: Willow, a free-text ACAA system; Willed, an authoring tool; Willoc, a configuration tool; and COMOV, a conceptual model viewer in which several representations have been used to show the conceptual models. In particular, these representations are: a concept map, a conceptual diagram, a table, a bar chart and a textual summary.

Resumen

Los profesores intentan transmitir su conocimiento a los estudiantes para que adquieran los conceptos a compartir por la comunidad. Sin embargo, lo que cada estudiante comprende puede ser muy distinto. Esto es, porque cada estudiante construye su propio modelo conceptual como una red de conceptos interrelacionados dependiendo de su bagaje particular. Además, según la teoría del aprendizaje significativo de Ausubel, para poder aprender nuevos conceptos es necesario tener unos conceptos previos con los que enlazarlos.

Los sistemas tutores inteligentes (ITS) son un tipo de software educativo que contiene un modelo sobre el estudiante y que se adaptan a su ritmo de aprendizaje. Los sistemas Hipermedia Adaptativos para la educación (HAE) están inspirados en los ITS para adaptar el contenido y la navegación en un curso al modelo de estudiante. Sin embargo, la evaluación de estos sistemas se limita a preguntas cerradas, que según la opinión de los expertos en evaluación no pueden identificar muchos de los errores conceptuales de los estudiantes. Por lo tanto, la evaluación automática de preguntas abiertas es un campo que se está estudiando bastante y que se basa en usar técnicas de Procesamiento del Lenguaje Natural (PLN). En la actualidad, existen alrededor de unos 20 sistemas de este tipo, pero ninguno usa un modelo de estudiante.

Se ha destacado la importancia de mantener un modelo del estudiante que sirva para identificar la falta de conocimiento previo y permitir que el sistema decida qué preguntas son las más adecuadas. Por lo tanto, en este trabajo, se han combinado ideas de los campos de PLN, HAE y evaluación de respuestas en texto libre para construir un puente entre lo que los profesores intentan transmitir en sus clases y lo que los estudiantes comprenden. Este objetivo se cumple mediante un nuevo procedimiento capaz de generar de forma automática los modelos conceptuales de los estudiantes a partir de sus respuestas a un sistema automático y adaptativo de respuestas en texto libre, que es la evolución de los sistemas automáticos de respuestas en texto libre incorporando el uso de un modelo de estudiante para adaptar la evaluación. Usando este procedimiento, es posible generar no sólo el modelo conceptual de cada estudiante, sino también el modelo conceptual de toda la clase.

De esta forma, los profesores pueden visualizar en cualquier momento el grado de asimilación de los conceptos expuestos en las lecciones y, discernir la falta de conocimiento para organizar más adecuadamente la agenda de sus cursos. Además, al usar el sistema automático y adaptativo de respuestas en texto libre, los estudiantes pueden conseguir feedback instantáneo y, al mirar a los modelos conceptuales, los estudiantes pueden organizar su estudio, revisando los conceptos qué ya han aprendido y los que aún les faltan o están conectados de forma incorrecta con otros conceptos.

El procecimiento se ha implementado en las herramientas Will que constan de: Willow, el sistema de evaluación automática y adaptativa de respuestas en texto libre; Willed, la herramienta de autor; Willoc, una herramienta de configuración; y COMOV, el visor de modelos conceptuales. En particular, cinco representaciones del modelo conceptual se han usado: un mapa conceptual, un diagrama conceptual, una tabla, un gráfico de barras y un resumen textual.

Acknowledgements

First of all, I would like to thank my tutors. I remember as it was yesterday when I finished the last exam of the Computer Science degree. I was only sure of one thing: I love teaching Computer Science. I had the opportunity of start working in a company, but the idea of waking every day without the satisfaction of doing what I enjoy most (to transmit knowledge to people who will become the next generation) made me feel too empty.

Therefore, I went to talk to one of my lecturers during the degree, Pilar Rodríguez. She is an excellent lecturer, researcher, manager and an extraordinary human being. She told me that in order to become a lecturer I had the possibility of starting the Ph.D. in the department and since then, she has always oriented me. Moreover, she has transmitted to me her joy for doing research on how to improve teaching with the use of the new technologies and introduced Enrique Alfonseca to me. I have no words to express my immense gratitude to Enrique. He had always been there. Whenever an article was accepted or when a problem appeared, I knew I could count on him. I cannot recall the many hours we have passed talking about this work. I will always be indebted to both of them.

I would also like to highlight the time I was in the ITC-irst research center in Trento (Italy). Bernardo Magnini, Carlo Strapparava, Luisa Bentivogli, Pamela Forner, Milen Kouylekov, Alfio Gliozzo and many more made me feel at home. Besides, I had the opportunity of working with many of them and learnt from their experience.

Many thanks to Manuel Alfonseca, Rosa Carro and Álvaro Ortigosa because they have helped me in many papers with useful comments and have provided me with the necessary data for the experiments. Furthermore, special thanks to Almudena Sierra, Manuel Cebrián, Eloy Anguiano and the students of Operating System subject of the Telecommunications Engineer degree because they have made true my dream of seeing the Will tools used by real students and how they have enjoyed it.

Thanks to Leila Shafti, Javier Bravo, Francisco Pérez, Manuel Freire, Estefanía Martín, Abraham Esquivel, Pedro Paredes, María Ruíz, Pablo Haya and the rest of the Computer Science department because they are the best colleagues I have ever had. Going everyday to the laboratory is nicer because they are there.

I cannot forget the many researchers I have had the opportunity of meeting in the conferences I have attended. From the very beginning, I have felt that I belong to a community and their advices have been a real engine for my research. In particular, the conversations with Vania Dimitrova, Jana Sukkarieh, John Oberlander, Dan Cristea and Kate Taylor as they have been very enriching.

My family is one of the most important aspects in my life, without their support I would have been unable to finish this work. I will never forget when for my first presentation, my sister Sonia Juana Pérez Marín acted as the audience, asking questions for a topic completely

different from her one. Big thanks also to Ismael Pascual Nieto, my soul mate, for giving me the strength to continue chapter after chapter. If I close my eyes, I can see him next to me during all the demos and experiments with the Will tools, always ready to help.

I dedicate this work to all of them because they are the underwriters of it.

Finally, many thanks to the Universidad Autónoma of Madrid because since I started my degree I have loved this place. To work for this university makes me feel the happiest person because I enjoy the work, the place and the people.

Acronyms

AC Area-of-knowledge Concept

ACAA Adaptive Computer Assisted Assessment

AEH Adaptive Educational Hypermedia

AH Adaptive Hypermedia

AI Artificial Intelligence

AR Anaphora Resolution

BC Basic concept

BLEU BiLingual Evaluation Understudy score

BN Bayesian Network

BP Brevity Penalty

CAA Computer Assisted Assessment

CAT Computer Adaptive Testing

CBR Case Based Reasoning

CMS Course Management System

COMOV Conceptual Model Viewer

CV Confidence Value

ERB Evaluating Responses with Bleu

ETS Educational Testing Service

FCA Formal Concept Analysis

FL Fuzzy Logic

GCSE General Certificate of Secondary Education

GMAT Graduate Management Admission Test

IE Information Extraction

IRT Information Response Theory

ITS Intelligent Tutoring System

LDA Latent Dirichlet Allocation

LFA Learning Factor Analysis

LSA Latent Semantic Analysis

MBP Modified Brevity Penalty

MCQ Multiple Choice Question

ML Machine Learning

MT Machine Translation

MUP Modified Unified Precision

NLP Natural Language Processing

NP Noun Phrase

POS Part Of Speech

PLSA Probabilistic Latent Semantic Analysis

RE Referential Expression

SAT Scholastic Aptitude Test

SVD Singular Value Decomposition

TC Topic Concept

TCT Text Categorization Technique

UM User Modeling

VSM Vector Space Model

WSD Word Sense Disambiguation

List of publications

This work has produced the following publications (most of them accessible on-line at [http1] and, related to the Chapters as indicated below):

- *Year 2003*:
  - **Automatic Multilingual Generation of On-line Information Sites**. Enrique Alfonseca, Diana Pérez and Pilar Rodríguez. In proceedings of the International Conference on Multimedia and Information Technologies for the Education (MICTE), Spain, November 2003. *Chapter 6, the Processing Module.*
- *Year 2004*:
  - **Application of the BLEU Method for Evaluating Free-text Answers in an E-learning Environment**. Diana Pérez, Enrique Alfonseca and Pilar Rodríguez. In proceedings of the International Conference on Resources and Language Evaluation (LREC), Portugal, May 2004. *Chapter 6, the Comparison Module.*
  - **Upper Bounds of the BLEU Algorithm applied to Assessing Student Essays**. Diana Pérez, Enrique Alfonseca and Pilar Rodríguez. In proceedings of the International Conference of the International Association for Educational Assessment, U.S.A, June 2004. *Chapter 6, the optimum use of Willow.*
  - **Automatic Evaluation of Users' Short Essays by using Statistical and shallow Natural Language Processing Techniques**. Diana Pérez under the supervision of Enrique Alfonseca and Pilar Rodríguez. Master Thesis presented at the Universidad Autónoma de Madrid, June 2004. *Chapters 6 and 8, the Comparison Module and the experiments with ERB.*
  - **Welkin: Automatic Generation of Adaptive Hypermedia Sites with NLP Techniques**. Enrique Alfonseca, Diana Pérez and Pilar Rodríguez. In proceedings of the International Conference of Web Engineering (ICWE), Germany, July 2004. *Another example of system using AH and NLP techniques.*
  - **Educational Adaptive Hypermedia meets Computer Assisted Assessment**. Enrique Alfonseca, Rosa María Carro, Manuel Freire, Álvaro Ortigosa, Diana Pérez and Pilar Rodríguez. In proceedings of the International Workshop A3EH of the International Conference in Adaptive Hypermedia (AH), The Netherlands, August 2004. *Chapter 6, the non-adaptive version: Atenea.*
  - **Automatic Assessment of open-ended Questions with a BLEU-inspired Algorithm and shallow NLP**. Enrique Alfonseca and Diana Pérez. LNCS 3230, Advances in Natural Language Processing, Springer Verlag, 25-35. Presented in the International Conference Spain for the Natural Language Processing (ESTAL), Spain, October 2004. *Chapter 8, the use of ERB and NLP for free-text assessment.*
- *Year 2005*:
  - **Application of the Bleu Algorithm for Recognising Textual Entailments**.

Diana Pérez and Enrique Alfonseca. In proceedings of the Pascal Challenges Workshop of textual entailment, U.K., April 2005. *Another application of ERB.*

- **Automatic Assessment of Students free-text Answers underpinned by the Combination of a BLEU-inspired algorithm and Latent Semantic Analysis**. Diana Pérez, Alfio Gliozzo, Carlo Strapparava, Enrique Alfonseca, Pilar Rodríguez and Bernardo Magnini. Published by the American Association for Artificial Intelligence (AAAI) Press. Presented at the 18th International Conference of the Florida Artificial Intelligence Society (FLAIRS), U.S.A., May 2005. *Chapter 8, the use of ERB and LSA.*
- **Adapting the Automatic Assessment of free-text Answers to the student**. Diana Pérez and Enrique Alfonseca. In proceedings of the 9th international conference on Computer Assisted Assessment (CAA), U.K, July 2005. *Chapter 5, the static component of the student model and the validation of using translated texts for the experiments.*
- **Authoring of Adaptive Computer Assisted Assessment of Free-text Answers**. Enrique Alfonseca, Rosa María Carro, Manuel Freire, Álvaro Ortigosa, Diana Pérez and Pilar Rodríguez. Journal of Educational Technology and Society, Special Issue on Authoring of Adaptive Hypermedia, International Forum of Educational Technology and Society, ISSN 1176-3647. Volume 8, Issue 3, July 2005. *Chapter 5, the authoring tool Willed.*
- **About the Effects of using Anaphora Resolution in Assessing free-text Student Answers**. Diana Pérez, Oana Postolache, Enrique Alfonseca, Dan Cristea and Pilar Rodríguez. In proceedings of the International Conference of Recent Advances in Natural Language Processing (RANLP), Bulgaria, September 2005. *Chapter 8, the use of Anaphora Resolution.*
- **About the Effects of Combining Latent Semantic Analysis with other Natural Language Processing Techniques to Assess open-ended Questions**. Diana Pérez, Alfio Gliozzo, Enrique Alfonseca, Carlo Strapparava, Bernardo Magnini and Pilar Rodríguez. Journal Signos. ISSN 0035-045 (press version), 0718-0934 (online version). Volume 38, number 59, December 2005. *Chapter 8, the use of ERB, LSA and NLP to assess free-text answers.*

- *Year 2006*:
  - **Using Bleu-like Algorithms for the Automatic Recognition of Entailment**. Diana Pérez and Enrique Alfonseca. LNCS 3944, Springer Verlag, Machine Learning Challenges, January 2006. *Another application of ERB and the wraetlic toolkit.*
  - **On the Dynamic Adaptation of Computer Assisted Assessment of free-text Answers**. Diana Pérez-Marín, Enrique Alfonseca and Pilar Rodríguez. LNCS 4108, Springer-Verlag. International Conference of Adaptive Hypermedia (AH), LNCS Springer Verlag, Ireland, June 2006. *Chapter 5, part of the dynamic component of the student model and about the synergy between AH and NLP techniques to be used together.*

- **A free-text Scoring System that Generates Conceptual Models of the Students Knowledge with the aid of Clarifying Questions**. Diana Pérez-Marín, Enrique Alfonseca and Pilar Rodríguez. In proceedings of the International WorkShop SWEL-AH, Ireland, June 2006. *Chapter 6, the clarification questions.*
- **Automatic Generation of Students' Conceptual Models underpinned by Free-Text Adaptive Computer Assisted Assessment**. Diana Pérez-Marín, Enrique Alfonseca, Manuel Freire, Pilar Rodríguez, José María Guirao and Antonio Moreno-Sandoval. In proceedings of the International Conference ICALT, IEEE, The Netherlands, July 2006. *Chapter 5, the procedure to automatically generate the students' conceptual models.*
- **Willow: Automatic and Adaptive Assessment of Students free-text Answers**. Diana Pérez-Marín, Ismael Pascual-Nieto, Enrique Alfonseca and Pilar Rodríguez. In proceedings of the International Conference of the Spanish Natural Language Processing Society (SEPLN), Spain, September 2006. *Chapters 6 and 7, a demo of the Will tools.*
- **Can Computers Assess open-ended Questions?**. Diana Pérez, Enrique Alfonseca and Pilar Rodríguez. Novatica journal, number 183, edited by the Association of Technics in Computer Science, September-October 2006. *Chapters 6 and 8, about the feasibility of free-text scoring.*
- **An Approach for Automatic Generation of Adaptive Hypermedia in Education with Multilingual Knowledge Discovery Techniques**. Enrique Alfonseca, Pilar Rodríguez and Diana Pérez. Computers and Education. In Press, Corrected Proof, available on-line from November 2006. *Another application of the combined use of Adaptive Hypermedia and Natural Language Processing techniques.*
- **Automatic Identification of Terms for the Generation of Students Concept Maps**. Diana Pérez-Marín, Ismael Pascual-Nieto, Enrique Alfonseca and Pilar Rodríguez. In the proceedings of the International Conference in Multimedia and Information Technologies for the Education (MICTE), Spain, November 2006. *Chapter 5, the Term Identification module.*

- *Year 2007*:
  - **Automatic Generation of Students' Conceptual Models from Answers in Plain Text**. Diana Pérez-Marín, Enrique Alfonseca, Pilar Rodríguez and Ismael Pascual-Nieto. To appear in the User Modeling 2007 International Conference proceedings, LNAI, Springer-Verlag, Greece, June 2007. *Chapters 7 and 8, COMOV and the first and second experiments with students.*
  - **Automatically Generated Inspectable Learning Models for Students**. Diana Pérez-Marín, Ismael Pascual-Nieto, Enrique Alfonseca and Pilar Rodríguez. To appear in the international conference Artificial Intelligence in Education (AIED) proceedings, IOS Press, U.S.A., July 2007. *Chapter 8, the third experiment with students.*
  - **A Study on the Impact of the Use of an Automatic and Adaptive free-text**

Assessment system during a University Course. Diana Pérez-Marín, Ismael Pascual-Nieto, Enrique Alfonseca, Eloy Anguiano and Pilar Rodríguez. To appear in the Blended Learning Pearson book, Edinburgh, U.K., August 2007. *Chapter 8, the third experiment.*

– **A Study on the Possibility of Automatically Estimating the Confidence Value of Students' Knowledge in Generated Conceptual Models**. Diana Pérez-Marín, Enrique Alfonseca, Pilar Rodríguez and Ismael Pascual-Nieto. To appear in the Journal of Computers (JCP), Academy Publishers, 2(4), July 2007. *Chapter 5, the procedure of automatically generating the conceptual models.*

Some explanatory notes

In this section, some explanatory notes to make the reading of this document easier are presented.

First of all, notice that through this work, the words teacher, instructor and lecturer will be used interchangeably. The same is applicable to the words student and learner.

Secondly, that *s/he* can be read as *she or he*.

Thirdly, the font type Sharif is used to attract readers attention to some important example that guides the reading of the theory presented in the main text.

Fourthly, all the examples mentioned in the text are related to the Operating Systems area. It is because this is the area in which I work as a teacher. However, the procedure described in the text is not restricted only to Operating Systems and could be applied to other areas provided that they there is not the necessity of assessing creative thinking or mathematical calculations, as will be explained in Chapter 9.

Fifthly, web references are numbered as [httpX] and gathered in the Web References section at the end of the document. The last date in which all of them were accessed was April the 15th.

Finally, it is important to highlight that although many of the Natural Language Processing techniques used during this work are applicable both to Spanish and English languages, the experiments with students have been done in Spanish in the Universidad Autónoma of Madrid since I work for this university.

Hence, while the results of the experiments that could be carried out without students (e.g. to measure Pearson correlations) are presented for English and Spanish languages, the experiments in which students were necessary, only the Spanish techniques have been applied to process the Spanish students' answers and generate the students' conceptual models. However, to make the reading of the chapters easier for non-Spanish speakers, the terms of these examples have been manually translated into English and they are shown in English in the main text, whereas Appendix E gathers the source terms in Spanish as they were used during the experiment.

Chapter 1

Introduction

Many philosophers, psychologists and researchers have examined the nature of knowledge, its possible classifications and representations. I would like to highlight the definition of knowledge given in Kang and Byun [2001] as it has been one of the sources of inspiration for this work: "*Knowledge is the product of a learning activity in which an individual assimilates and accommodates new information into his or her cognitive structure in accordance with the environment as s/he understands it.*"

That is, a constructivist view of knowledge is followed in this work, according to which we construct knowledge of reality that fits our experience as the result of our interactions with the world [Carpendale, 1997]. In particular, each student builds his or her specific cognitive structure or conceptual model (understood as a network of concepts) depending on his or her particular features and previous knowledge and thus, the way in which each student is able to understand the lesson may be quite different. Moreover, according to the Meaningful Learning Theory of Ausubel et al. [1978], each student will only be able to learn new concepts if s/he finds the necessary previous concepts to link the new ones in his or her conceptual model.

However, it is commonly agreed that the use of conventional tests is not enough to build the bridge over the gap between what students actually learn and what they should understand as a consequence of instruction. There are studies in the literature that report cases in which students with high scores in objective tests (Multiple Choice Questions, fill-in-the-blank,...), when interviewed by a teacher, have shown deep underlying misconceptions [Sigel, 1999].

Therefore, it is necessary to have some reliable strategy to model the student's conceptual knowledge. In this work, a new procedure is proposed able to automatically generate students' conceptual models from free-text answers and, show them to teachers and students. This procedure is based on the synergical combination of techniques from Natural Language Processing and Adaptive Hypermedia fields, and it has been implemented in the **Will tools**, which consist of the following systems (all of them accessible on-line for academic use):

- **Willow**, a free-text Adaptive Computer Assisted Assessment (ACAA) system [http2].
- **Willed**, an authoring tool [http3].
- **Willoc**, a configuration tool [http4].
- **COMOV**, a conceptual model viewer [http5].

1.1 Goals

The main goal of this work is to **build a bridge over the gap between what students actually understand and what the teachers think that they have transmitted during their lessons**. The motivation is not just to put an automatic score to the students or to give more feedback to them, but to provide the teachers with the possibility of having instant access to the fleeting conceptual model of each student, and to have the possibility of automatically generating the conceptual model of the class, which is very hard to do by hand.

This main goal encompasses the following subgoals:

- **For teachers:**
  - To provide them with feedback to know how well the concepts taught have been understood and to identify misunderstandings.
  - To allow them to keep track of each student and the whole class learning progress, i.e. to find out how the new concepts modify the previous ones and to discover the new links that are being created.
- **For students:**
  - To provide them with a procedure that assesses their answers in free-text in an adaptive and automatic way, able to show them instant feedback for each question answered.
  - To allow them to identify their main misconceptions by having a look at some representation of their conceptual model and the whole class conceptual model.
  - To foster reflective thinking in the students by guiding them to the correct answer with a set of clarification questions automatically generated by the free-text Adaptive Computer Assisted Assessment (ACAA) system.
- **Others:**
  - To overcome the limitations imposed by traditional objective testing sections in Adaptive Educational Hypermedia systems by allowing students to answer open-ended questions.
  - To make the assessment of free-text answers adaptive so that it is guided by a student model (that is modified by the assessment outcomes) and using techniques not only from Natural Language Processing but also from Adaptive Hypermedia.
  - To find out the best combination of Natural Language Processing techniques in order to improve the accuracy of the automatic assessment of the answers written in free-text by the students (without need of training or asking teachers to fill in complex templates).

1.2 A sample scenario

In order to illustrate the aforementioned goals in a practical example, let us suppose that we have a group of students enrolled in the Operating Systems subject of the Computer Engineering degree. They are compelled to attend lessons, as their final score will not only depend on the

grade of their exam but also their work during the semester (an example of continuous evaluation system).

The teacher of the subject, Juan, follows a complicated agenda in which some lessons strongly depend on previous ones. He would like to know if the preceding concepts have been understood so that he can continue with the next ones. Thus, after teaching something new, he always asks whether the previous lesson has been understood. The problem is that he always receives as answer the silence of the whole class. He would like to do more exams during the semester but he does not have time to review all of them. He would also like to do more practical exercises in class but, he has no time if he wants to finish the theoretical part of the subject.

This scenario is the ideal case to use an automatic and adaptive free-text scoring system. It is a web application so that no student has to install it. Besides, no computer knowledge (apart from being able to use a web browser) is necessary to use it. In fact, Juan only has to provide the system with the set of questions to ask and the correct answers using an authoring tool that has been developed to make this task easier (it is also an on-line application, which does not require any computer knowledge).

Next, he should include as one of the activities of the continuous evaluation the use of Willow some hours per week. Juan's students can access the system in the laboratories of the faculty or from their home. The only requisite is that the computer has Internet access.

All Juan's students are given an account when they finish the registration process (also on-line). In the registration they are asked their name, age and language (necessary for their profile). Their name and password will be used to access the system from anywhere and any time. For each particular session, they can choose how many questions they want to answer, how long they want to practise, the topics and even the level of feedback. Next, they only have to answer the questions asked by the system, according to their profile, which is being updated as they keep answering. Besides, a log system is recording all their actions.

Whenever a student fails a question, a clarification set of questions starts in order to guide him or her toward the correct answer, that is, the solution to the question is not immediately given (to foster reflective thinking). Finally, when the question is passed or the compensation questions have all been failed, the automatic score, the processed answer and the correct answers given by Juan are shown to the student. Non-passed questions will be asked again in the future.

This way, Juan's students can benefit from immediate feedback and see whether they are understanding the lessons and what they should review (formative assessment). They know that the more they practise, the more opportunities they have of passing the final exam. Moreover, they can organize their study so that they can focus on the concepts which have not already been assimilated.

On the other hand, Juan fulfills his wish of being able to provide more training to his students without having to review all the extra homework. Even better, he has instant access to each student's conceptual model and the whole class conceptual model with a conceptual model viewer such as COMOV. That way, he can see which concepts have been worst understood as his students do not use them in their answers to the system and which ones have already been assimilated. He can also detect misconceptions (concepts wrongly connected) and lack

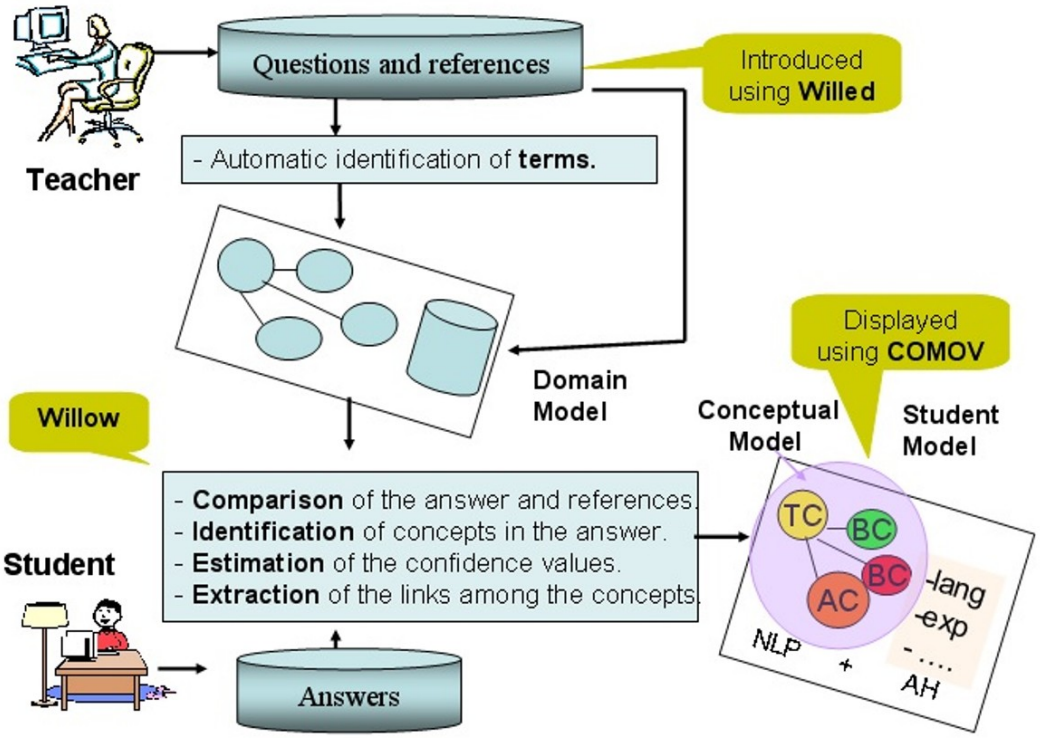


Figure 1.1: Picture of the approach proposed in this work.

of concepts (concepts missing). Therefore, Juan is given the feedback he needs to adjust the agenda of the course and avoid starting a new topic until the previous one has been understood (at least by the majority of the class).

1.3 Thesis overview

1.3.1 General idea

Figure 1.1 gives a general view of the approach proposed to automatically generate the students' conceptual models. As can be seen, the teacher is asked to use an authoring tool to introduce the questions and its correct answers (references) in the database. The references are automatically processed to generate the domain model.

Next, whenever a student answers one of the questions proposed by the free-text ACAA system, not only s/he gets instant feedback but, his or her use of the terms of the domain model is analyzed to generate his or her student model. The student model consists of personal data gathered from the student and the generated conceptual model.

Finally, the conceptual model can be shown to teachers and students with a conceptual model viewer to identify which concepts of the lessons should be reviewed and, which ones have already been assimilated.

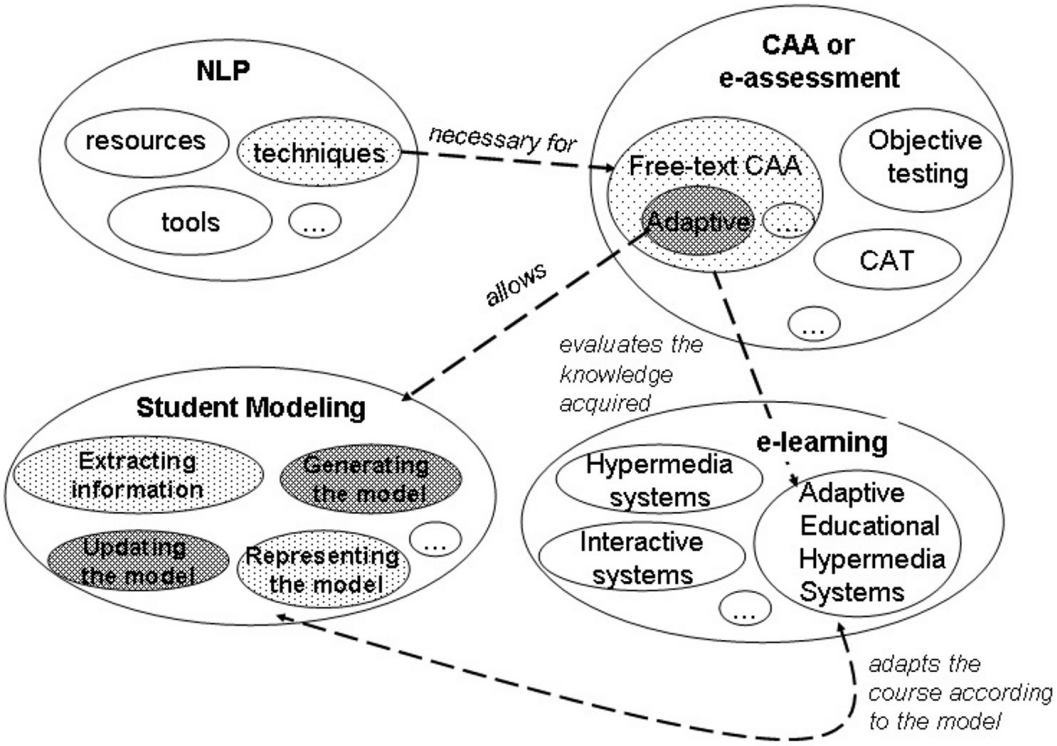


Figure 1.2: Fields to which this work is related and how they are related among themselves. Notice that the most shadowed fields are the ones that have been more addressed.

1.3.2 Context

It is important to highlight that given the interdisciplinary nature of the goals pursued, this work is not related only to one field but to several related fields as can be seen in Figure 1.2.

In particular, it is related to:

- **Computer Assisted Assessment or e-assessment** because this field studies how to effectively use computers to assess student's learning.
- **Student Modeling** because this field studies how to model students so that the information gathered in the models can be used as feedback to the teachers, students or internally by the system.
- **Adaptive Educational Hypermedia** because this field studies the techniques to take into account each student's model and act accordingly.
- **Natural Language Processing** because this field studies how to automatically process free-text and thus, from this field, the techniques to automatically assess free-text answers are retrieved.

1.3.3 Organization

This work is organized in three main parts:

- **State-of-the-art review**: It comprises Chapters 2-4 and presents the theoretical foundations of this work and the review of the state-of-the-art of the main fields to which it

has made a contribution:

  - *Chapter 2* gathers several cognitive and pedagogic theories on which this work is based. That is, it validates that a mental model can be inferred and expressed as a conceptual model. Moreover, that it is possible to represent this conceptual model in several knowledge representation formats such as a concept map.
  - *Chapter 3* presents what student's modeling is and, in particular, the conceptual modeling subset. The topics covered are not only to define conceptual models, but also to review how they are being built and represented. Some systems based on conceptual models are also presented as related work.
  - *Chapter 4* gives an overview of the state-of-art of CAA for free-text answers. In particular, it provides a review of the current statistical, Natural Language Processing and other techniques that are being employed, and how they are used by the currently available automatic free-text scoring systems.

- **Proposal**: Chapters 5-7 describe a procedure for automatically generating the students' conceptual models, as well as some systems that implement it.
  - *Chapter 5* describes the procedure to generate the students' conceptual models. Besides, it gives a step-by-step illustrative example of the procedure based on the scenario presented in Section 1.2.
  - *Chapter 6* details the architecture of the free-text ACAA system called Willow. The modules are described one by one and also how they work integrated. Other related developed software in the framework of the Will tools will also be explained such as Willed (the authoring tool) and Willoc (the configuration tool).
  - *Chapter 7* introduces COMOV whose goal is to display the generated conceptual models to teachers and students in five different knowledge representation formats: a concept map, a conceptual diagram, a table, a bar chart and a textual summary.
- **Experiments and conclusions**: Chapters 8 and 9 end with the explanation of the experiments performed, their results, the conclusions drawn and future work.
  - *Chapter 8* validates the feasibility of the procedure to automatically generate the students' conceptual models from the point of view of the teachers and the students. Moreover, it describes the design and results of the experiments performed which prove that the goals gathered in Section 1.1 have been fulfilled.
  - *Chapter 9* sums up the main conclusions of this work and how it can be further extended. It also gathers the necessary information to apply the procedure to different language and/or area-of-knowledge.

Additionally, this work has five appendixes: Appendix A presents the technical details of this work; Appendix B gives some extra information about the experiments; Appendix C is the Spanish version of Chapter 1; Appendix D is the Spanish version of Chapter 9; and, Appendix E provides the terms in Spanish used for the experiments and manually translated into English to make the reading of Chapter 7 easier for non-Spanish speakers.

Part I

State-of-the-art review

In this part, the theoretical foundations of the work proposed are established.

It consists of three chapters:

- Chapter 2 entitled "**Some cognitive and pedagogic theories**" reviews the most important cognitive and pedagogic theories that support the idea of using students' conceptual models as feedback both for teachers and students to identify which concepts have been assimilated and which ones should be reviewed. Moreover, to represent the organization of concepts from the internal structures in students' minds to a processable format.

- Chapter 3 entitled "**Students' conceptual modeling**" gives answers to questions such as what conceptual models are, why they are relevant and how to build and represent conceptual models. This Chapter ends by gathering some systems underpinned by the use of conceptual models.

- Chapter 4 entitled "**Computer Assisted Assessment of free-text answers**" is dedicated to the field of automatic assessment of free-text answers. It reviews the techniques, evaluation procedures, metrics and systems that are being used. Finally, it presents a comparison of these systems.

Chapter 2

Some cognitive and pedagogic theories

This section explores some cognitive and pedagogic theories to support the idea that people think in terms of concepts. Moreover, that people express these concepts and their relationships in language when learning or being assessed. One of the most powerful and widely used representation form of conceptual structures is concept maps. They are useful for many different educational applications both for teachers and students.

For instance, let the following dialogue, which is a modified fragment of the original in Dimitrova [2002], serve as a motivation sample to show how the mental cognitive structure of a student can be used to guide the assessment. Concepts are marked in bold and, as can be seen, the goal of the dialogue is to find out which concepts are missing and help the teacher to clarify them:

TEACHER Have you heard of **operating systems**?

LEARNER Yes, I think so. Isn't this **Windows**?

TEACHER **Windows** is one example of an **operating system**. Do you know any other **operating systems**?

LEARNER No. I thought **operating system** meant **Windows** and am confused now. What is an **operating system**?

TEACHER **Operating systems** are **computer programs** that maintain the communication between a **computer** and the **programs** that run on it. **Windows** is one example of an **operating system**. **Linux** is another example of an **operating system**.

2.1 Definition of concepts and their categories

It is commonly agreed that knowledge, in part, is expressed in concepts. People use concepts in their daily lives. But what is a concept? How are concepts structured in our mind? How can

we model these relationships? These questions have been addressed for many researchers since the fifties.

Bruner was one of the first researchers working with concepts. He devised the **Classical Theory of concepts** according to which concepts are complex mental representations whose structure generally encodes a specification of necessary and sufficient conditions for their own application. Bruner stated that concepts have "defining features" [Bruner et al., 1956].

However, this early theory was still very simple as it was unable to explain how some concepts share features with others. For instance, a "*tomato*" has features belonging both to "*fruit*" and "*vegetable*". Another critic that the Classical Theory received is that it did not include the "*typicality effects*". In particular, Rosch [1978] claimed that individuals are more influenced by what they think that is more typical than by common features, when they are requested to categorize concepts. Error rates show that the more typical X is to the target category Y, the fewer errors of categorization. Besides, Rosch [1978] also claimed that when subjects provide exemplars of a given category they cite first the more typical items.

Therefore, in the seventies, the **Prototype Theory of concepts** was developed in order to include the typicality effects. This theory gave up the idea that the concept's internal structure provides its definition. It adopted a probabilistic treatment of conceptual structure that replaced the "defining features" idea with "common features" per category of concepts (concepts with similar features). In fact, this theory added the notion of "family resemblance". Thus, for the example, a "*tomato*" is a vegetable because although it has some features of the fruit category of concepts, it has more similarities (more family resemblance) with the features of vegetables.

Nevertheless, two main problems were also identified for this theory:

- **Prototypes are incomplete**: Prototypes are not valid even with something as simple as a tomato. It is commonly assumed that a tomato is red and round. Thus, provided that the exemplar of tomato considered is red and round there is no problem, but if the tomato is still green, then it does not work, as a tomato should be red according to its prototype.
- **Some complex concepts do not have prototype**: For instance, there is no prototype for grandmothers whose grandchildren are computer engineers.

In the eighties, Novak defined a concept as "*a perceived regularity in events or objects, or records of events or objects, designated by a label*". The label can be one or more words [Novak and Gowin, 1984]. From this definition, two important features of a concept can be identified:

- **Its archetype nature**: It does not refer to something concrete, but to a regularity in a set of objects. For instance, a concept is not a particular computer but what we, human beings, understand as a computer: a machine that process information with a CPU and some input and output devices.
- **Its necessity of being denominated**: It needs to be assigned a label, something that identifies it since without this label the concept is inaccessible.

It is also important to observe that some concepts are more based on superficial features than others. For example, stereotypes are superficial but commonly used to express personal knowledge of people and the social world [Kunda, 1999]. On the other hand, scientific concepts,

which have contributed to the advancement of knowledge, are based on more essential features [Lewin, 1969].

Another interesting topic to mention is categories. They serve to group concepts with similar features. In particular, the variability of a category is a direct function of the number of properties which are different, and an inverse function of the degree of similarity of the members with respect to the similar properties [Tversky, 1977]. At least two methods have been reported to decide how much variability is acceptable within a category before splitting it into sub-categories (see, for example, Safayeni et al. [2003]):

- **Calculate the similarity among members**: It could be done by comparing the features of each concept with their prototype [Rosch, 1978, Taylor, 1995] or as proposed by Medin et al. [1993] by relying on a measure of perceived similarity among the concepts in the category.
- **Consider categorization as a part of a larger system of classification**: According to the Gestalt theory there are two forces that interact in order to determine the variability. One of them aims to unify similar concepts into the same category and the other aims to separate them. At the end, what happens, is that these forces can be quantified as the ratio of perceived variability within categories to the perceived variability between categories. In this way, the overall distribution is restricted to a number of categories that is manageable and coherent with a bigger taxonomy.

Cognitive psychology has eventually rejected the classical view of concepts and, it is considered that there are different levels of abstraction for concepts. For example, the concept "*justice*" is more abstract and different than object-based concepts such as "*dogs and boats*" [Medin et al., 2000]. In fact, at the highest levels of abstraction, concepts may not necessarily be categorized neither be represented by exemplars. In any case, in this work, it is going to be followed the Novak's definition of concept.

2.2 From concepts to words

Murphy [2002] has suggested that concepts can be considered as the non-linguistic representations of classes of entities, while words are labels that map these concepts onto our knowledge structure. As it has been seen in the previous section, concepts need to be named in order to make them accessible. Thus, words are a system used to describe and to name concepts. However, naming a concept increases its variability of meanings [Safayeni et al., 2003]. For instance, if we consider the concept of "*game*", we can analyze all its different instances as it is done in Wittgenstein [1958].

It is also important to highlight that not all words serve to convey concepts since some of them express actions or links. In this work, a special relevance will be given to the words that express concepts (usually nouns or group of nouns) and their label will be called a term as in Rovira [2005]. In fact, a term is usually defined as a word or a multi-word expression that is used in specific domains with a specific meaning.

In education, terms have been used to describe knowledge domains and for concept maps. To

describe knowledge domains, a common vocabulary is established and then the interrelationships between these domain concepts form the domain structure. In this way, the deep conceptual knowledge of the domain is presented and it is also possible to annotate learning resources improving their findability, shareability and reusability [Dicheva and Dichev, 2004].

Term extraction is an important problem in the Natural Language Processing (NLP) area [Cabré et al., 2001]. Proposed solutions to term extraction usually analyze large collection of domain-specific texts and compare them to general-purpose text, in order to find domain-specific regularities which indicate that a particular word or multi-word expression is a relevant term in that domain. Term candidates are usually returned ranked according to some specific metric or weight that indicates its relevancy. Several techniques have been devised to identify and extract nominal terms of a text:

1. Statistical corpus-based approaches such as the ones devised by Pantel and Dekang [2001] and Drouin [2003].
2. Linguistic processing techniques such as part-of-speech patterns, or the use of parsers [Bourigault, 1992, Voutilainen, 1993, Eklund and Wille, 1998].
3. Hybrid approaches which combine statistical techniques and linguistic knowledge [Justeson and Katz, 1996, Maynard and Ananiadou, 2000].

2.3 Relationships among concepts

The relationships between concepts may be static or dynamic. Both systems are considered as necessary for representing knowledge. Static relationships connect the concepts in propositions (basic unit of representations that join two concepts with a relationship that is stated on the link between concepts). They help to describe, define and organize knowledge for a given domain. In this way, they reduce the uncertainty in the labels. Besides, they are considered as a means of organizing scientific knowledge in a hierarchical form. At least three different types have been reported [Jonassen, 2000, Safayeni et al., 2003]:

- **Subsumption**: When a concept $C1$ is a subclass of another concept $C2$. For example, squares ($C1$) are a subclass of geometric shapes ($C2$).
- **Common membership**: When two concepts $C1$ and $C2$ are together a subclass of another concept $C3$. For example, squares ($C1$) and triangles ($C2$) are both a subclass of geometric shapes ($C3$) and thus, they are related.
- **Intersection**: When a new concept $C3$ is generated by crossing two sources concepts $C1$ and $C2$. For example, symmetrical geometric shapes ($C3$) is generated from symmetrical shapes ($C1$) and geometric shapes ($C2$).

Dynamic relationships reflect the effect of a change in one concept on another. They are concerned with co-variation among the concepts. For example, $C1$ causes, changes or influences concept $C2$. These relationships are viewed as means of representing scientific knowledge about how change in one concept affects another one. Two different types of relationships have been reported according to which they are based [Thagard, 1992]:

- **Causality**: When a concept $C1$ causes another concept $C2$. For example, a certain virus ($C1$) causes a certain illness ($C2$).
- **Correlation / probability**: When a concept $C1$ is directly or inversely related to another concept $C2$. For example, infection with Ebola ($C1$) is a good predictor of death ($C2$) [CDCP].

2.4 Mental models

According to Johnson-Laird [1983], mental models are internal representations of situations in the world that are expressed by word referents. They consist of knowledge that "plays a direct representational role since it is analogous to the structure of the corresponding state of affairs in the world - as we perceive it". Other definition of mental models labels them as dynamic and runnable systems experienced as imagery [Glasser et al., 1987].

At least three different mental representation theories have been reported [Sigel, 1999]:

- **Empiricist**: Mental representations are just passive reproductions caused by the effects of the world on the mind like a camera or a fossil.
- **Constructivist**: People construct knowledge of reality as a result of their interactions with the world.
- **Correspondence theory of knowledge**: Knowledge consists in having a mental representation that corresponds to reality as it really is.

Concepts have no constant structure but are continually created [Jones and Smith, 1993] (i.e. the mental models have a fleeting nature) and are organized into hierarchies, at least for individuals older than six years old [Inherlder and Piaget, 1964]. Johnson-Laird [1983] argued that building mental models requires placing concepts in the context of a large system of knowledge and relating them to other concepts.

This inner flexibility of representations may cause that, from the source information, each individual acquires something that could be completely different. This has been called by some researchers as "a plague on attempts to educate and evaluate" [Sigel, 1999]. In fact, some studies report that what learners actually learn and what they should understand as a consequence of instruction is often very different.

A possible way to discern faulty or incomplete knowledge is by performing alternative assessment based on concept maps. Not only by directly asking people to create the concept maps as sometimes they are not consciously aware of their knowledge or able to express what they know but by other means [Sigel, 1999]. For instance, according to the Theory of Performance, which characterizes domain understanding, evidence and tasks in term of performance expectation and certainty, it is possible to claim that "given (high,low) understanding of concept C, it is (more, less) certain that someone should produce evidence Y when performing task Z" [Gitomer et al., 1995]. That is, from our internal concepts organization, a mental model which justifies certain actions or expressions can be created.

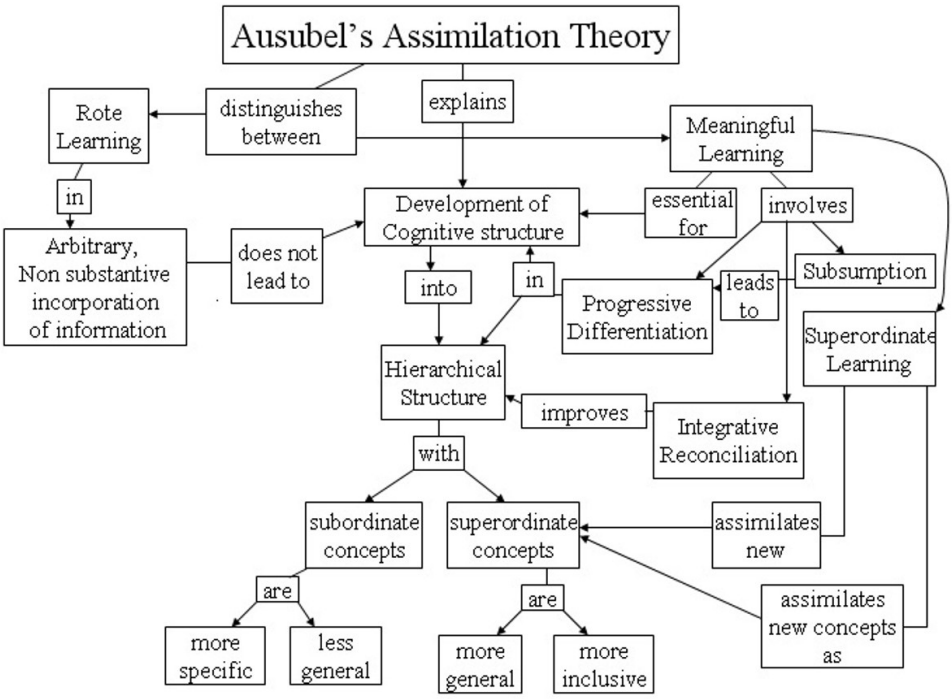


Figure 2.1: Overview of the Ausubel's meaningful learning theory.

2.5 Ausubel's Meaningful Learning Theory

The American psychologist Ausubel devised a learning theory called the Meaningful Learning Theory whose main premise is that the acquisition of new knowledge is dependent on what is already known [Ausubel et al., 1978]. This theory has been successfully used in education during the last decades. See Figure 2.1 for an overview of the theory.

According to this theory, concepts are objects, events, situations or properties that possess common criteria attributes and are designated by some sign or symbol. Two methods of concept learning can be distinguished: concept formation and concept assimilation. Concept formation is for young children that acquire the criteria attributes of the concepts through direct experience. While concept assimilation (the predominant method for concept learning) is for school children and adults that actively incorporate new knowledge by linking it to previous one (anchorage of new information to existing ideas). In other words, construction of knowledge begins with our observation and recognition of events and objects through concepts we already possess.

Cognitive theorists believe that it is essential to relate new knowledge to existing information. Teachers can facilitate learning by organizing information presented so that new concepts are easily relatable to concepts already learned. When this happens, and students link new knowledge with relevant, preexisting concepts or propositions in the students' cognitive structure, it can be said that there has been meaningful learning. In this way, the new concepts can be better retained and applied.

Meaningful learning is personal, idiosyncratic and involves a recognition of the links between concepts [Ausubel, 1963]. To learn meaningfully, students must intentionally attempt to

integrate new concepts with existing ones so that they can interact in the learner's knowledge structure. In this way, they get a more extensive network of knowledge (conceptual model) and more retrieval paths.

The opposite of meaningful learning is rote learning which also can incorporate new information into the knowledge structure but without interaction with previous knowledge. It occurs when the student just memorizes new data without creating any link to previous knowledge. It is fine for lower cognitive levels, such as remembering sequences but does not aid the learner in understanding the relationships between the objects. Therefore, according to Ausubel, two type of learners can be distinguished: rote learners who just memorize data and meaningful learners who meaningfully learn. Besides, rote learners have less extensive networks and less retrieval paths between knowledge concepts than meaningful learners.

Both rote and meaningful learning may be achieved no matter what instructional strategy is used [Novak and Gowin, 1984]. However, meaningful learning should be promoted as it allows students to go for higher cognitive levels and improve the management of the concepts of a domain. In order to achieve meaningful learning both reception learning (a teacher speaking about a topic he has chosen and passive students listening) or discovery learning (the student chooses what to learn) can be employed. That is, in order to meaningfully learn, how the information is presented is irrelevant, the point is how the information is integrated into the existing knowledge structure. Here, teachers should promote students to think about their previous concepts and their interactions with the new ones. They should not be given data just to memorize them but to investigate the relationships between the concepts therein.

Finally, it is important to highlight that, according to Ausubel, the depth of concepts is different. That is, concepts can range from the very general to the very specific. General concepts subsume less general concepts which include most specific concepts. As such, concepts can be progressively differentiated by their level of specificity. Thus, learners who aim to learn meaningfully must discern the level of new concepts and place them within progressively inclusive levels of specificity in their knowledge structure.

2.6 Novak's concept maps as a tool of Ausubel's theory

The theory of Meaningful Learning is the fundamental pillar of concept maps, useful and powerful tools to visually represent the conceptual structure that someone has about an area of knowledge. Novak introduced them as a tool for students to freely organize their knowledge about a certain area [Novak, 1977, Novak and Gowin, 1984]. A concept map is a way to model mental knowledge [Chen et al., 2005]. In this way, they bring to light individual differences in learning, as different people have different concept maps, even on the same content, and none of them can be cited as the only correct one.

Concept maps were originally designed as an education aid to assist students in organizing the concepts in a limited domain by connecting them with labeled links. They have been widely used on many levels of education from elementary school to university studies as an aid to help people conceptualize and organize their knowledge not only for learning but also for assessment

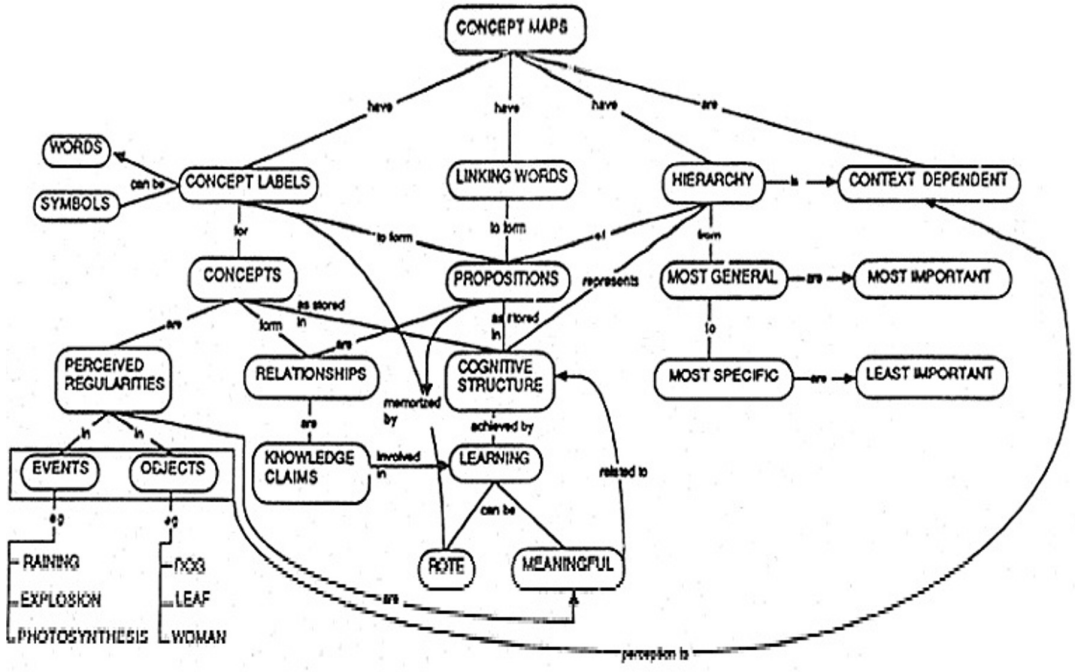


Figure 2.2: An example of Novak's concept map (source: Novak et al., 1984).

[Novak and Gowin, 1984, Plotnick, 1997, Hsu et al., 2005, Sormo, 2005].

This representation takes the form of a graph or a diagram that shows the concepts and the connections that students have between them [Geller, 2004, Caas et al., 2003]. Three main basic elements can be identified in a concept map (see Figure 2.2 for an example):

- **The concepts**: They are represented in the graph as the nodes. In a concept map, each concept only appears once.
- **The links**: They join two related nodes. A node can be related to one or more nodes. Arrow symbols are used at the end of each link to describe the direction of each relationship.
- **The propositions**: As it has been seen in Section 2.3, they are the basic unit of meaning. They are created from the composition of the labels of the concepts and the label of their link that indicates the type of relationships between these nodes.

The organization of knowledge in a concept map can be hierarchical (with the main concept at the top and less relevant concepts below to show super- and sub-ordinate relationships) and/or linear (with concepts placed in a cluster or network patterns) [West et al., 1991]. In fact, Jones et al. [1987] categorized concept maps according to how the concepts are related in three different types: spider, chain and hierarchy map.

West et al. [1991] also added hybrids of these three basic types like a hierarchy map that has a spider map as a part of it. In general, irrespectively of the organizational structure and the area of knowledge under consideration, the more meaningful connections a person can show in the concept map, the better s/he understands the material.

Concept maps can make clear to the student and the instructor how small is the number of truly important concepts they have to learn. Voluminous prose can be distilled into essential and linked ideas. In addition, since concept maps externalize a person's knowledge structure then concept maps can serve to point out any conceptual misconceptions the person may have concerning the knowledge structure. This explicit evaluation of knowledge and subsequent recognition of misconceptions allows for finely targeted remediation. Furthermore, since concept maps are visual images they tend to be more easily remembered than text.

Concept mapping has a variety of applications within a broad range of domains. In fact, they have been very extensively used in several fields (sciences, politics, statistics, business...) and by very different users. For instance, automatic diagnosis of diseases in expert systems, student guidance in e-learning courses, and in traditional education, for all ages and phases, from the design of the lessons to their evaluation. Besides, concept maps require the learner to operate at all six levels of Bloom's taxonomy [Bloom, 1956].

Therefore, concept maps are one way to foster and measure meaningful learning in the classroom as instructional, student learning, and assessment techniques. By using concept maps, the structure of a student's knowledge, the validity of proposed concepts, the existence of misconceptions (when erroneous propositions are found) and the lack of previous knowledge (when two concepts that should be linked are isolated) can be analyzed [Ruiz-Primo and Shavelson, 1996]. This can help teachers to make their instruction more effective [Ross and Munby, 1991].

Three main purposes of concept maps can be distinguished according to Zimmaro and Cawley [1998]:

- **Instructional tools**: Concept maps could be used by instructors of the same department or institution to facilitate knowledge management by organizing course content, presenting material to students and communicating complex ideas.
- **Student learning tools**: Concept maps help students to learn about their knowledge structure and the process of knowledge construction (metaknowledge) and to learn how to learn (metalearning). In particular, they facilitate knowledge acquisition by reconciling new and old knowledge, encouraging the students to generate their own connections between concepts and integrating material across different courses.
- **Student assessment tools**: Concept maps can be used to evaluate student learning. Furthermore, to see if and how meaningful learning is occurring. They facilitate knowledge assessment by relying on them to create the questions, giving feedback to students and instructors and evaluating the end of course knowledge.

  However, when using concept maps for assessment purposes, it is important to realize that the experience of the student in doing concepts maps may influence the quality of the resulting map. Less experienced students could produce lower quality maps even having a good understanding of the material. It would be helpful to check the ability of the student in creating concept maps as well as the knowledge exposed in the maps or to find alternatives such as automatically generating the concept maps according to free-text answers as is proposed in this work.

Therefore, concept maps can be used not only as learning or metacognitive tools but also as

tools for evaluating knowledge representations and structures [Novak and Gowin, 1984, Novak, 1990, Mintzes et al., 2000, Olea, 2006]. More about concept maps and their underlying theory can be found in Caas et al. [2003] and Novak and Canas [2006].

2.7 Ausubel and Novak ideas in practise

Web-based distance education has rapidly become very popular. As well as in traditional teaching, it is crucial to determine for each course what to teach, how to teach and how to ensure learners' mastery of the material. Good human instructors can intuitively make these determinations, whereas computers must be programmed as Intelligent Tutoring Systems (ITSs) or Adaptive Educational Hypermedia Systems (AEHs). It is usually done with three different components: expert model, student model and instructor model [Burns and Capps, 1988, Shute and Torreano, 2002].

These three models jointly specify "what to teach and how to teach it" as the instructor model determines how to ensure learner mastery by monitoring the student model in relation to the expert model and addressing discrepancies. However, there are still some unsolved problems to effectively manage courses, most of which are brought by the difficulty to gain sufficient understanding of social, cognitive, and behavioral aspects of distance students [Smith-Gratto, 1999, Helic et al., 2000].

Students may feel disoriented without the support of a tutor [Cotton, 1988, Hara, 2000]. The regular monitoring of the students' behavior can reduce these problems [Galusha, 1997, Cotton, 1988, Ragan, 1999]. Besides, educators demand to be able to have an overview of the performance of their students, monitor discussions, cluster learner groups based on certain patterns of behaviour/performance, identify tendencies in different groups and discover common misconceptions (not only for traditional students but also for distance ones). Hence, the instructors should be provided with tools to keep track of the students' models and so, to be able to prevent or overcome potential conflicts [Rueda et al., 2004, Mazza and Dimitrova, 2005].

Concept maps are particularly useful for representing the networks of concepts in students' minds. In particular, Horton et al. [1993] performed a study in which the effectiveness of concept mapping for improving students' achievement and students' attitudes was analyzed. The results indicated that concept mapping raised student achievement on the average by 0.46 standard deviations, as well as a strong improvement in student attitude.

Therefore, it could be concluded that concept maps should be the common knowledge representation tool today and they have extensively used in the classrooms. However, it is not the case. It could be due to the fact that they are time consuming to learn how to create them and difficult to manage in paper [Kremer, 1994, Hsu et al., 2005]. To solve this problem and thanks to the generalization of computers in education, many computer applications have been developed to support the creation and maintenance of concept maps. Automated tools can improve visual appearance and consistency. For instance, with programs such as CMapTools [Caas et al., 1999] or CMTool [Rocha and Favero, 2004].

Chapter 3

Students' conceptual modeling

A model can be defined as a simplified representation of the real world. That is, an imitation of a particular phenomenon of the real world in a smaller scale, but preserving all the details, necessary to facilitate its apprehension. Models have been used for many different applications such as testing a theory, predicting economical investment evolutions or explaining someone's thought. According to Jameson [1999], when describing a user model, three aspects have to be considered: the information that it includes and how it is obtained, the representation of this information in the system and, the process of forming and updating the model.

The application of student models for education is considered in this Chapter. Student or learner modeling is concerned with the task of keeping a record of many aspects of a student. This record is called a student model and it may comprise: domain-specific information such as how much and what the student has learned to date, his or her misconceptions and problem solving strategies; and, learner-specific characteristics such as the student's learning style or affective dimensions.

According to Hartley and Mitrovic [2001], student modeling may be characterized as:

- **Passive**: The educational system infers the model of the students without explicit help from them.
- **Active**: Students may be asked questions by the educational system to assist it.
- **Interactive**: Students play an active role in the development and maintenance of their own model.

Finally, Jameson [1999] has divided the information about the student into: static properties, that cover the student's personal characteristics, capabilities, preferences, etc. which are constant through the learning process; and, dynamic properties, that involve information about the student interaction with the system and thus, they change during the learning process. This covers student's knowledge, concepts and skills, learning style, motivation, viewpoints, current goals, plans and beliefs, learning activities that have been carried out, objectives that have been achieved, etc. As this information changes, the student model should be updated accordingly.

3.1 Points of view about students' models

Students' models can be discussed according to many different points of view. In this section, four of the most relevant ones are reviewed. They focus on: the goal pursued, the degree of openness, the relationship between domain and student knowledge and, the granularity of the model.

Firstly, depending on the task to which the model is going to be used, four possibilities can be distinguished [Frias-Martinez et al., 2004]:

- **Content-based filtering or prediction**: It is based on the assumption that the future will be similar to the past. Thus, by relying on past behavior, the student model can be used to predict future needs. For instance, if a student has reviewed the topic of scheduling, it is marked as interesting in his or her model as it is expected that s/he will be interested in more information about this topic in the future.
- **Collaborative filtering or recommendation**: The student model is used to suggest which elements may be interesting to the student according to the opinions of other students when they were suggested those elements. For instance, if other students have shown their interest in scheduling, then it is also marked as interesting for this student.
- **Classification**: The model is used to assign one of several predefined classes to some items. For instance, if the student is classified as an advanced learner, more difficult items are selected.
- **Detection of misconceptions and lack of knowledge**: To warn teachers and students that some concepts should be reviewed as they have been misunderstood or they are unknown.

Secondly, regarding their openness, models can be shown to instructors, students, both or any of them [Hartley and Mitrovic, 2002]:

- **Open**: They are shown to students or instructors. In this way, students may get actively involved in their diagnostic process by looking at how they are understanding the concepts in the learning domain. Besides, educators can be provided with more feedback about their students' knowledge assimilation state and help them to improve it [Bull and Nghiem, 2002, Dimitrova, 2002, Rueda et al., 2004]. They are used in active and interactive student modeling. They can be subclassified as [Bull et al., 2003]:
  - **Inspectable**: They show the model to instructors and/or students but they do not allow them to modify it.
  - **Editable**: The model is built and kept by the educational system but it allows instructors and/or students to modify its contents.
  - **Negotiated**: The model is agreed between the educational system and the student that freely interacts with it (e.g. through a dialogue). Generally, this model is not aimed to be shown to the instructor as it is less oriented to blended learning (i.e. combined used of the computer and human tutors).
- **Closed**: They are shown neither to instructors nor to students as their aim is just to modify the behavior of the educational system to be adapted to the student. They are

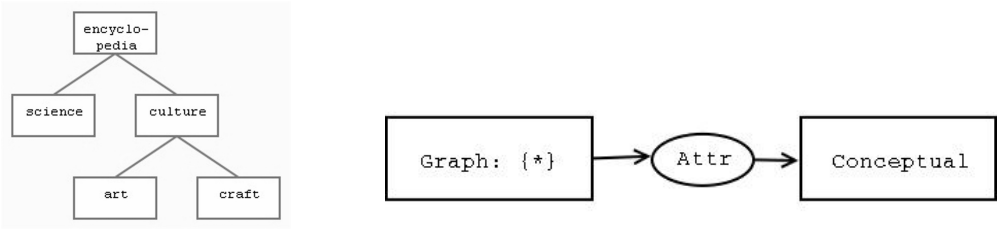


Figure 3.1: An example of a tree structure on the left and of conceptual graph on the right.

used in passive student modeling.

Thirdly, concerning the relationship between domain and student knowledge. That is, depending on how they represent the knowledge domain regarding the student knowledge [Labidi and Sergio, 2000, Mitrovic, 2001]:

- **Overlay**: The student model is a projection of the domain model, i.e. the student knowledge is considered as a subset of the domain knowledge.
- **Bug**: The bug model is based on a library of possible mistakes that could be made up by the student in its pedagogical activities.
- **Perturbation**: The perturbation model is an hybrid model that involves the concepts of the overlay and bug together.
- **Constraint-based**: Opposite to the previous models, it does not compare the student's knowledge to the domain knowledge. It rather focuses on correct knowledge by checking if all the constraints of a certain domain are satisfied by the student.

Finally, according to the granularity of the model, i.e. the number of students modeled [Gouli et al., 2004]:

- **One student**: The main goal of the model is to capture information from a particular student.
- **A group of students**: The model represents general information about a group of students. It can also be distinguished according to how they are built from:
  - **Models of individuals**: It is directly done by somehow combining the models of particular students that have been created with a similar structure to make easier their composition.
  - **Scratch**: It does not rely on the particular students' models. As the models of individuals, it is created from the interactions of the students with the system. It also may be done collaboratively [Frias-Martinez et al., 2004].

3.2 Representation forms of students' conceptual models

In the last decades, many different forms of knowledge representations have been devised. Some of them are: tree structures, predicate logic, concept maps, conceptual graphs, frames and rules, self-organizing feature maps, ontologies, tables, bayesian networks, semantic networks, topic maps and textual summaries.

The aim in this section is not to review all of them in detail but just to focus on three

Name	Goal	Structure	Flexibility	CL?
Tree structures	Classify items	Hierarchical	Low	No
Conceptual graphs	Represent first-order predicate logic	Graph	Medium/High	No
Concept maps	Represent cognitive structures	Graph	High	Yes

Table 3.1: Comparison table between three different knowledge representation forms.

of them: tree structure, conceptual graph and concept maps. They have been chosen because they have been applied for educational purposes to foster some of the cognitive and pedagogic theories explained in the previous Chapter. The representation forms are:

- **Tree structures**: In graph theory, a tree is a connected acyclic graph that permits to represent the hierarchical nature of a structure. It is named "tree structure" because the graph looks like a tree with a starting node that is the "root" placed at the upper edge and from it several other ones (the children). The lines connecting the nodes are the "branches". When a node has no other node parting from it is called a "leaf". There is no connection between nodes at the same level that are called "brothers" or "sisters". See Figure 3.1 left for an example of tree structure.
- **Conceptual graphs**: They are a graphic notation for typed first-order predicate logic devised by Sowa [1984]. They serve as an intermediate language between the computer and the humans by expressing meaning in a formal but readable format. They are similar to concept maps as they are also graphs with nodes and links between them. However, they are not the same as they are bipartite graphs. That is, there cannot be links between a concept and another concept or between a relation and another relation. In fact, the structure is different as in a concept map the nodes are always concepts and the arcs express the relationships between these nodes with a label attached to it. In a conceptual graph, the nodes can represent concepts or relationships and the arcs can only connect a concept node with a relation node. An example of conceptual graph can be seen in Figure 3.1 right. The same conceptual graph can also be represented linearly as: `[Graph: *]->(Attr)->[Conceptual]`.
- **Concept maps**: As it has been seen in Section 2.6, concept maps are a visual tool to represent as a graph the conceptual organization of someone about an area of knowledge.

Table 3.1 presents a comparison among them according to their main goal, type of structure, grade of flexibility and if they allow cross links (CL?, long connections between concepts in different domains).

3.3 Methods to build and update student models

The task of creating and keeping a good student model (in particular, a good student's conceptual model) is a complicated task. In order to deal with it, several methods have been proposed and used. In this section, some of them are reviewed.

The simplest method to get the student's model is just to directly ask students to provide

the necessary information. Initially, it can be done by requesting some forms to be filled by the students with the properties to be modeled and after that by letting them to update the model via web or mail. For instance, the traditional procedure to create a concept map by hand according to Novak and Gowin [1984] is:

1. **Identify the key concepts**: It can be done by instructors or by the students. When the list of key concepts is generated by instructors, in some cases it is possible to allow students to add their own concepts. The instructors usually choose these key concepts by looking at paragraphs, research reports, or chapters, or just by thinking of the concepts of a subject area and listing them [Sims-Knight et al., 2004, Ruiz-Primo, 2004]. However, there are some critics to this approach, as leaving the decision to humans make it subjective [Leake et al., 2004] and two humans tend not to agree completely. When the list is generated by students, it is important to make sure that they have enough experience to correctly choose these terms.
2. **Write and arrange the concept labels (terms)**: Some people find helpful to write each term on a separate card or small piece of paper, so that they can be moved around. All the same, it is only necessary to write them as the nodes of the graph in a pattern that best represents the information. Hierarchical or non-hierarchical structures can be used. In the case of hierarchical maps, the concepts could be ranked by placing the broadest and most inclusive idea at the top of the map. Non-hierarchical maps permit to put concepts any place near related concepts.
3. **Connect the concept nodes by lines**: Each concept can appear only once and it should be enclosed in a circle or oval. Next, lines with arrows (single or double-headed) should be drawn to capture each relationship between two concepts. Each line is associated to a label consisting of one or more linking words. These linking words together with the terms of the concepts joined can be read as a proposition. There is no limit to the number of links stemming from any node. Finally, cross links between concepts in different sections of the concept map can be created.

On the other hand, several techniques have been devised to approach the student modeling task (building and updating) automatically by using Machine Learning (ML). These techniques usually work in two phases: training and testing. During training, from a set of cases, some parameters are tuned so that the algorithm can apply them in the test with new cases. In particular, so that from the interaction of the students with the educational system, the ML technique can construct their models. Some of them are:

- **Case-based reasoning (CBR)**: Originally, Intelligent Tutoring Systems (ITS) had a set of prototypical student profiles and then, they asked students in order to assign them their more related prototype [Wenger, 1987]. CBR offers an improvement to that approach as it gives information about the student from a set of stored case bases. For instance, an ITS for helping students to solve exercises would have stored a set of solved problems. Thus, when the student is asked to solve an exercise, his or her solution is compared against the previously stored cases and according to how similar they are, the ITS can follow with more or less difficult problems [Sormo, 2005].

- **Bayesian networks (BN)**: They are a form of probabilistic graphical model. They can be represented by a directed acyclic graph whose nodes are variables and arcs are statistical dependence relations among them. They offer solid theoretical foundation, are consistent and powerful in reasoning, but need a detailed model description together with estimations of all parameters. Therefore, as some researchers such as Millan-Valldeperas [2000] have claimed, this technique is computationally complex, and difficult to implement.
- **Fuzzy logic (FL)**: As the determination of student knowledge can be very imprecise and approximate, and may include a great deal of uncertainty, the fuzzy set theory can be chosen to deal with vagueness in the model [Zadeh, 1965]. In particular, fuzzy logic defines a framework to capture and deal with the ambiguity of the information. Traditionally, a fuzzy logic inference system consists of three phases: (1) fuzzification; (2) fuzzy inference; and (3) defuzzification. However, for student modeling the inference process is restricted as the goal usually is recommendation and filtering. Its implementation is simple and computationally undemanding. On the other hand, its main disadvantage is the need of building a fuzzy rules database and the sometimes unpredictable behavior of the model [Kavcic, 2001].
- **Clustering**: In this technique, each data point belongs only to one cluster and, the distance between the clusters to represent the student interactions with the educational system can be given by different algorithms. A special case of clustering, is fuzzy clustering in which, with a certain degree, the data points can belong to more than one cluster and, an example of algorithm for calculating the distance between the clusters is Fuzzy C-Means [Bezdek, 1981]. Fuzzy clustering is usually applied for recommendation and classification tasks.
- **Neural Networks**: They are an information processing paradigm inspired in how biological nervous systems such as the brain extract complex patterns from data. Traditionally, they have been used for classification and to recommend the next step for a given student trajectory in a virtual environment. However, they can also be used for filtering and prediction.
- **Genetic Algorithms**: They are inspired in the mechanisms of natural selection to make the search of the parameters of the model more effective. The initial step is to create a population and then, to make it to evolve until it is closer to the optimum solution to the problem at hand [Holland, 1992]. In general, they have been used for recommendation in the form of rules, which can capture user goals and preferences, because they perform a global search and cope better with attribute interaction than other algorithms where the search is more local.

Between the manual and the fully automatic techniques, it should also be cited the negotiation techniques. Their core idea is that the system together with the student build the student model in an iterative process. Finally, it is important to highlight that a combination of two or more techniques is frequently applied. This assures more accurate modeling and allows better exploitation of the information gathered. For instance, fuzzy logic has been combined with Machine Learning to produce behavior models that attempt to capture and to manage

Name	Automatic?	Goal	Granularity	Complexity
By hand	No	D.	I	Med/High
Case-based reasoning	Yes	C	I	Med
Bayesian networks	Yes	P+C+R	I+G	Med/High
Fuzzy logic	Yes	F	I+G	Med
Neural networks	Yes	C+R	I+G	High
Genetic algorithms	Yes	R	G	High
Fuzzy clustering	Yes	C+R	G	Med/High
Neuro-fuzzy	Yes	P+C+R	I+G	High

Table 3.2: Comparison table between different techniques to build students' models. In the goal column: P. indicates prediction, R. recommendation, C. classification and D. detection and in the granularity column: I means for modeling individuals and G for modeling groups.

the uncertainty of human behavior [Frias-Martinez et al., 2004] and fuzzy systems have been combined with neural networks to produce neuro-fuzzy systems that are basically fuzzy logic systems with an automatic learning process provided by the neural network [Magoulas et al., 2001].

The main advantage of the simple approach of modeling by hand is that the information gathered is highly reliable as it has been directly given from the students. However, the disadvantages are many: they require students to learn how to build the concept maps (it might be quite time-consuming) and, some students may not have expertise enough to be reliable in the process of building a concept map that really reflects their knowledge. Besides, in the case that concept maps are drawn in paper, their management might be thorny for a great number of students and to update the model, it is necessary to repeat the whole process.

On the other hand, the automatic techniques while trying to keep the benefits can remedy the problems of manual approaches. In particular, sometimes to make the updating process easier additional techniques are used such as: Learning Factor Analysis (LFA), that is a semi-automated method for improving cognitive models and is based on a combination of a statistical model, human expertise and a combinatorial search [Cen et al., 2006]; model refinement, that is based on theory refinement that starts with an initial knowledge base and keeps correcting errors in the knowledge base from error examples until the knowledge base is consistent with the examples [Baffes and Mooney, 1995]; and, Q-matrix that automatically extracts features in the problem set to discover knowledge structure from the student response data [Barnes, 2005].

Table 3.2 shows a comparison of the main techniques discussed in this section. It can be seen that no technique is ideal for all situations and, how they address different goals.

3.4 Some systems underpinned by conceptual models

In this section, some of the most recent and relevant systems that are underpinned by the ideas of Ausubel and Novak explained in Chapter 2 are presented.

ALE [Kravcik and Specht, 2004] is an adaptive and adaptable learning environment that

provides individualized education in the area of design and architecture. Its main goal is to foster meaningful and multidisciplinary learning. It also allows discovery learning by providing concept-based navigation. Moreover, it keeps a model of the students with information about their learning style to adjust the navigation possibilities to them and supports coaching by relying on case-based reasoning. The student's model always reflects the current state of the student's progress to give the most suitable recommendations based on the student's learning style, preferences and knowledge stored in the model.

COMPASS [Puntambekar et al., 2003] is an AEHS that supports the assessment as well as the learning process by displaying each concept on a separate page and allowing the user to navigate the concepts using textual hyperlinks as well as a clickable concept map. COMPASS student models are very simple only based on their navigation behavior: which concepts they have visited and in what order. These student models can represent individuals or groups.

ConceptLab [Zapata-Rivera and Greer, 2001] is a knowledge construction and navigation system that uses XML-based concept maps to represent the student's view of the domain. It has three main goals: to assess the student's knowledge (it can be done by comparing different maps visually or through queries), to determine problems in the learning process of a student or a group of them and to promote reflection among a group of students in a topic. The student model is based on a bayesian network and a concept map. According to the authors, the concept map has been included as part of the student model in order to facilitate sharing of knowledge among students and assessment of students' knowledge by teachers. The concept maps are collaboratively built by the students who can be helped by a guide concept map. By clicking on a particular concept, it is possible to access a variety of links, added by the teacher or classmates, related to the concept of interest.

The knowledge built with ConceptLab can be represented with the **VisMod** system [Zapata-Rivera, 2004]. VisMod allows students and teachers to experiment with the creation of Bayesian what-if scenarios; providing not only a visualization tool, but also an interactive tool for inspection of and reflection on Bayesian student models.

The **CREEK-Tutor** [Sormo, 2005] is an ITS focused on helping students to solve exercises in areas such as mathematics, computer programming and medical diagnosis. The system asks the students to draw concept maps to develop a more accurate student model and to link the exercises to the textbook knowledge. The concept maps are compared to predict the difficulty of the exercises. In this way, the CREEK-Tutor can suggest the appropriate difficulty level exercises. Moreover, it can suggest next steps in the reasoning of an exercise by matching the current state of problem solving with the reasoning traces performed by previous or prototypical students solving the same exercise and showing the zone of the concept maps that supports the recommendation given.

DynMap+ [Rueda et al., 2004] is a graphical tool to display the student model as a concept map. Students introduce the concept map in the computer using the Concept Map Editor provided. DynMap+ can show models not only of individuals but also of groups. Both are overlay models that can be shown to students and instructors. The purpose of showing it to instructors is to provide them with a view of the knowledge and evolution of the students.

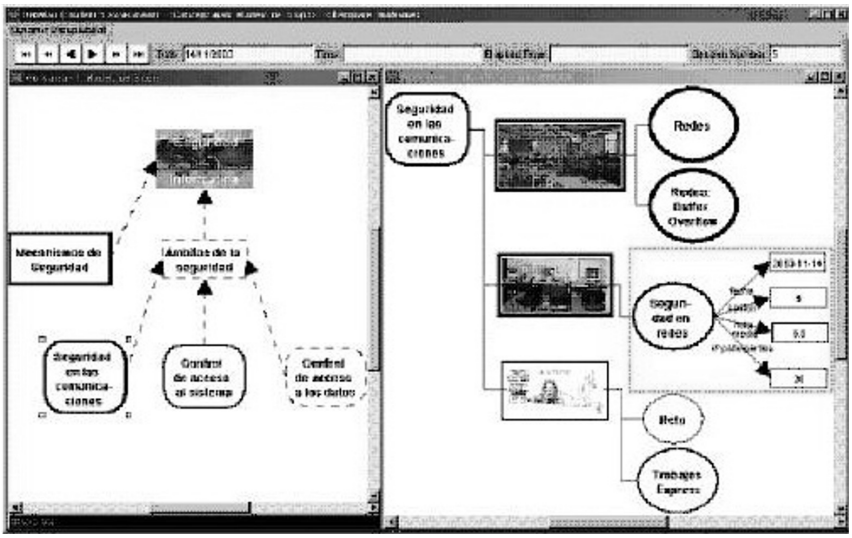


Figure 3.2: Snapshot of Dynmap.

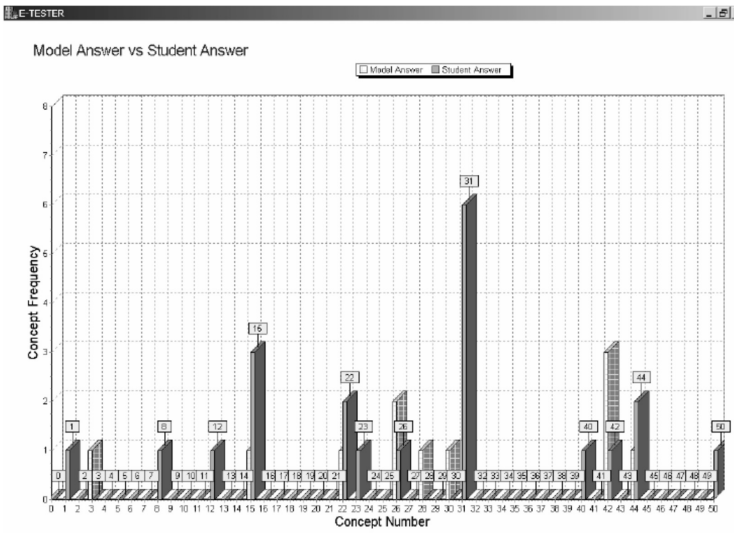


Figure 3.3: Snapshot of an E-TESTER output.

The purpose of showing the map to students is to foster reflective thinking about their own learning. See Figure 3.2 for a snapshot of the system.

E-TESTER [Guetl et al., 2005] is a computer-based system that identifies the main concepts in a text and generates questions from these concepts such as "What is *xxx*?" or "Explain *yyy*". Next, it waits for the students' answers in free-text to compare them with the e-learning content that the system has and treats as model answer. The comparison is based on the free-text scoring system Markit (described in Section 4.3.15). The difference is that in E-TESTER the process is simplified as it is only focused on counting the frequencies of the identified concepts in the student's answer and in the model answers. In fact, its feedback is not a numerical score but a visual representation of the student strong and weak points. See Figure 3.3 for an example of E-TESTER output.

It can be seen how each bar in the graph refers to an identified concept. Naturally, the more similar they are, the better the knowledge the student has about the concept. Whenever a student answer contains concepts that do not appear in the model answer, it is said that the student has introduced irrelevancies into the answer (e.g. concept number 31). The opposite case in which the model answer contains concepts that do not appear in the student answer is called "ignorance" or deficit of knowledge (e.g. concept number 3).

KBS Hyperbook [Henze et al., 1999] is an Educational Open Adaptive Hypermedia tool based on the use of an adaptive hyperbook. Adaptive hyperbooks are information repositories for accessing distributed information gathered from several web pages irrespectively of their origin (this is why they are open systems), integrated and adapted to a particular user's needs (this is why they are adaptive). It is based on the constructivist theory as it fosters that students do not act as passive consumers of information but actively link new information to previous data and discuss content to construct their own knowledge.

A typical snapshot of KBS Hyperbook system has, on the left page, the index of concepts to review (each one with a color circle icon indicating the degree of study completeness) and, on the right page, there is the information about a topic. This way, students can read as with a traditional book, but with the advantage of having more powerful navigation possibilities thanks to the links and the color system, which indicates them what they should review. For instance, students are warned not to follow links marked with red color circle icons because they do not have enough knowledge to understand their content.

LEO [Coffey, 2005] is a system based on the Ausubel's Meaningful learning theory that provides students with a graphical schema of the course, links to instructional content and a visual representation of their progress. LEO is part of a the "CMapTools" software suite that allows experts to structure knowledge of a domain as a concept map. In particular, it is the organizer of the suite, that is, the tool that provides the framework to organize all the information associated to the course as an augmented concept map in which the nodes can contain the instructional topics to learn (topic nodes) or additional information regarding the topics (explanation nodes) and the links create many potential paths through the material.

LEO presents both a global and a local view of the course structure. It also allows students to choose which subsets of the concept map should be shown. Topic nodes have color codings at the left to indicate student progress through the course of instruction and the links between topic nodes convey prerequisite relationships among topics. Possible status for a topic node are: completed when the instructor sees that the student has done enough work for a topic to be completed; ready, when previous required information has been completed and thus, the topic node can be studied; not ready, when previous required information has not been completed and thus, the topic node cannot be studied yet; and, current to indicate the current topic node under study. As the student works through the course, the model is being updated so that the next time that the student logs into the system s/he can see the updated map.

STyLE-OLM [Dimitrova, 2003] is a diagnostic tool integrated in the STyLE educational system that interactively builds the student model through a dialogue based on conceptual graphs between the student and the system. Its main goal is to engage students in reflective

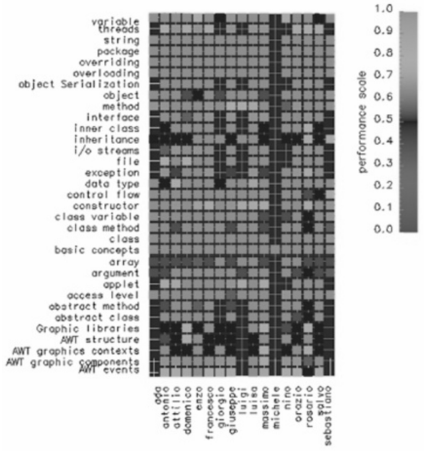


Figure 3.4: A snapshot of the CourseVis system.

activities. Instructors can use a specifically designed module to create the domain model of the course as a list of domain concepts for each page of the course and for every question in the quizzes. The domain model is stored as XML and introduced as an ontology into the system. The student model is an overlay of the domain model which incorporates the student's beliefs and misconceptions formalized with Prolog rules based on conceptual graphs. Beliefs can be correct (supported by the domain ontology), incomplete (facts from the domain ontology that the student does not believe and can be elicited by the system or stated by the student with an "*I do not know answer*" to a system's question) or erroneous (not supported by the domain ontology), and are open for inspection and discussion.

The resulting negotiated student model can be visually depicted with **CourseVis** as can be seen in Figure 3.4. It shows a cognitive matrix in which the students are mapped onto the x-axis and the concepts of the course are mapped onto the y-axis. The performance values are mapped onto the color of the square corresponding to a student and a concept. This matrix is shown to instructors so that they can detect problematic topics, struggle students by comparing columns and row or analyze the performance of a particular student on a topic. It is also possible to generate other views of the student model such as a student's histogram with information about how s/he has accessed the content pages of STyLE by topics, global access to the course, progress with the course schedule, messages and assignment submission. Some deficiencies found when evaluating CourseVis conducted to a new version of the system called **GISMO**. It includes representation of the dependencies among the concepts and is integrated in the Moodle [http6] open source learning environment.

TADV [Kosba et al., 2005] is a computer-based advice generating framework designed to deliver advice to instructors and facilitators in web-based distance education environments. It consists in two parts: the first one represents the conventional structure of a course and the second one is to model students and generate advice. Regarding the first part, the course designers are responsible for preparing the course material that is usually organized hierarchically. That is, a course is a set of lessons and, each lesson has a set of concepts which comprise the knowledge building blocks. Each concept is illustrated by learning objects (HTML pages, pre-

Name	Goal	Domain
ALE	Allow discovery learning	Architecture and design
COMPASS	Support learning and assessment	General science
ConceptLab+VisMod	Knowledge construction and visualization	Biology
CREEK-Tutor	Exercise-oriented tutoring solving	Factual domains
DynMap+	Graphically visualize student models	Computer Science
E-Tester	Diagnose student's knowledge	Computer Science and Law
KBS Hyperbook	Guide the learning of students	Computer Science
LEO	Show a visual representation of the knowledge	General science
STyLE-OLM+GISMO	Diagnostic tool	Terminology
TADV	Give advice to instructors	Mathematics

Table 3.3: First comparison table between ten different systems underpinned by conceptual models.

sentations, etc.), assessment quizzes, and communication activities. TADV uses concept maps to represent the relations between domain concepts in a hierarchical structure.

The second part is in charge of keeping track of the students while they navigate through the course and store the collected data in a database. It is done not only for each particular student (student model), but for certain groups (group model) and the whole class (class model). These models together with the information of the concept maps are used to produce advice to instructors about possible problems and needs of individuals and groups of students, as well as to suggest appropriate actions, when possible. For instance, if a student's knowledge status of a concept is marked as unlearned, the instructor could send a mail to the student to tell him to review that concept.

These systems are only a sample. There are many other educational systems that are underpinned by some kind of conceptual model. For example, in Hwang [2003] they propose a system for modeling prerequisite relationships among concepts to be learned and in this way providing appropriate individual learning guidance to enhance learning performance. In Seta et al. [1997], the Conceptual Level End-user Programming Environment (CLEPE) system keeps a table of the relationships among terms. In Muehlenbrock et al. [2005], the Monitor system maintains a knowledge model of the learner based on a knowledge level for the indicated relevant concepts. In Leake et al. [2004], they use the concepts to look information in the Internet. Dicheva and Dichev [2004] use terms to describe a domain and Carlson and Tanimoto [2005] induce text classification rules from student answer sets to construct diagnoses of misconceptions, which teachers can inspect to monitor the progress of their students and, to automatically construct formative feedback.

3.5 Comparison and conclusions

This section presents two comparison tables of the ten systems described in the previous section according to following items:

- **Goal**: Main purpose of the system.

Name	Language	Type	Addressee	KRF	Technique
ALE	English	Inspectable	Student+Instructor	Tree	CBR
COMPASS	English	Closed	Program	Graph	PFNets
ConceptLab+VisMod	Spanish.	Negotiated	Student+Instructor	Concept map	BN
CREEK-Tutor	English	Closed	Program	Concept map	CBR
DynMap+	Spanish	Inspectable	Student+Instructor	Concept map	By hand
E-Tester	English	Inspectable	Student+Instructor	Histogram	Statistical
KBS Hyperbook	German	Inspectable	Student	Ontology	By hand
LEO	English	Inspectable	Student+Instructor	Concept map	By hand
STyLE-OLM+GISMO	English	Negotiated	Student+Instructor	Conceptual graph	Logic
TADV	English	Closed	Program	Fuzzy variables	Fuzzy Logic

Table 3.4: Second comparison table between ten different systems underpinned by conceptual models.

- **Domain**: In which they have mainly been used. It does not mean that they cannot be applied to other domains as well.
- **Language**: The language in which the environment has been created and is able to process.
- **Type**: The type of model, i.e. closed, inspectable, or negotiated.
- **Addressee**: Target role or roles of the addressee(s) of the conceptual model.
- **KRF**: The knowledge representation form.
- **Technique**: The technique used to build the student model.

Each table gathers each item in a column and each system in a row. Table 3.3 focuses on the goal and domain of each system whereas Table 3.4 shows the data about the rest of the items. It can be seen that there have not been restrictions in the domain of application. These systems have been applied from factual domains to numerical and textual domains. All the same, Computer Science is one of the most popular domain fields. This could be because the authors of these systems are usually instructors of Computer Science subjects. Only two systems allowed negotiated student models, being the most popular option to have an inspectable model to show both to students and instructors. There is no technique that can be stated as the best one or a best knowledge representation form. It is also important to highlight that only one of the systems reviewed (E-TESTER) apply Natural Language Processing techniques.

Finally, it can be concluded from the experiments carried out to evaluate these systems that irrespectively of the country, level of education to which they were applied and technique used, instructors showed their satisfaction with gaining more feedback about their students' knowledge. Instructors mostly appreciated the possibility of not only having individual students' models but also group models. Furthermore, students did not have any problem using (even helping to build) concept maps and, appreciated to get more feedback and be given the possibility of keeping track of their evolution as consequence of their learning progress.

Chapter 4

Computer Assisted Assessment of free-text answers

Teachers all over the world spend a great deal of time just marking students' works. Hence, they have to cut down the time they can devote to their other duties. Even doing that, sometimes they do not have enough time to properly assess the works of the big number of students they have. Therefore, many authors believe that this situation has to be solved and, some of them have presented the computer as a new assessing tool able to automatically assess free-text answers. Most of these authors do not attempt to substitute the teacher with the computer, but to help the teachers with the computer software. Besides, although the technical approaches that these free-text Computer Assisted Assessment (CAA) tools are undertaking are very different, the goal and concepts underlying are common for all of them as it will be shown in this chapter.

Bloom [1956] provided a taxonomy for categorizing the level of abstraction of questions used in the assessment of student work. He identified six different levels: knowledge, understanding, application, analysis, synthesis and evaluation. This taxonomy has been taken as the starting point for analyzing the student's learning competence. Table 4.1 summarizes the main features of each competence level. The relevant assessment methods to be employed in each level are stated according to Bishop [2002].

Many authors agree that multiple-choice questions only serve to evaluate the lower levels in the taxonomy. When it is necessary to measure the higher levels, open ended questions should be employed [Birenbaum et al., 1992, Mcgrath, 2003, Mitchell et al., 2003, Palmer and Richardson, 2003].

On the other hand, the automated assessment of students' essays is regarded by many as the Holy Grail of Computer Assisted Assessment (CAA) [Whittingdon and Hunt, 1999]. There has always been hard critics about the idea of a computer grading human essays. Nowadays, there are still some skeptical researchers that do not consider that the automatic grading is possible. However, the advances in NLP, Machine Learning and neural network techniques, the lack of time to give students appropriate feedback (despite the general assumption of its importance) and the conviction that multiple-choice questions cannot be the only assessment method are favoring a change in this situation. See Figure 4.1 to have a glimpse of the many

Competence	Question cues	Skill demonstrated	Assessment methods
Knowledge	List and define	Recall of information	A quiz
Understanding	Summarize and predict	Grasp the text meaning	An exam
Application	Illustrate and solve	Practical use of the material	A role play
Analysis	Infer and compare	Notice patterns and hidden data	Analyze a computer output
Synthesis	Integrate and rewrite	Digest information	Open ended questions
Evaluation	Recommend and grade	Judge value for purpose	Peer assessment

Table 4.1: Review of the six Bloom's competence levels, the main skill that they demonstrate, two examples of question cues and a relevant assessment method for each of them (source: Bishop, 2002).

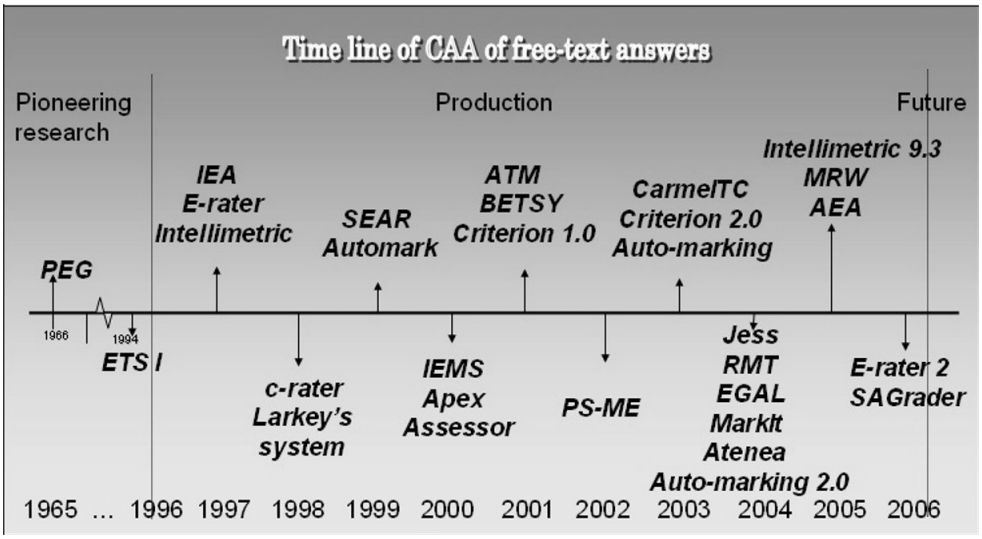


Figure 4.1: Time line of research in CAA for free-text answers.

existing free-text CAA systems and their dates of appearance.

The key question is how a computer can effectively measure the student knowledge. Page [1966] made a dichotomy between evaluating content and style. Content would refer to what the text says, and style refers to syntax, mechanics, diction and other features of the writing. Some researchers are strongly against this classification because they think that, in order to assess the text, both content and style are important, and one should not consider the former without the latter [Christie, 2003].

In order to grade the style, some CAA software tools look for direct features in the text, such as word number, word lengths or use of adjectives, and translate them into more abstract measures such as variety, fluency or quality. Although there are some critics to this procedure [Hearst, 2000], it is widely accepted and it can provide good results [Christie, 2003].

Concerning essay content evaluation, several automated assessment techniques have appeared recently, and some of them are even commercially available. Besides, several traditional tests such as the Graduate Management Admissions Test (GMAT), the Test of English as a Foreign Language (TOEFL) or the Graduate Record Examination (GRE) are including open ended questions with a computer-based delivery, which may support the use of automated

scoring methods.

Moreover, if the instruction is aimed to be adaptive, it is first necessary to capture the student's understanding of the subject [Sellers, 1998]. Knowledge assessment is the key to tailoring the learning experience, so it is an essential component of any adaptive teaching system. A good assessment can be used to make decisions about the teaching process [Dietel et al., 1991]. Without assessment, there is no way to measure the results of teaching, or tailor further education [Barnes and Bitzer, 2002, Coffey, 2005]. Besides, the enhancement of the assessment process with adaptive capabilities is worthwhile for at least two reasons [Gouli et al., 2002]:

- To make it dynamic and individualized as it is adapted to the student's performance.
- To reduce the number of questions required to estimate the student's knowledge level.

However, the current attempts of making CAA systems adaptive have been limited to: adapt the problem selection [Mitrovic and Martin, 2004], the navigation through the problems [Gutiérrez et al., 2004, Sosnovsky, 2004], the feedback provided to the students [Lutticke, 2004] or, Computer Adaptive Testing (CAT) [Wainer, 2000] that modifies the order in which the test items are presented to the students according to their performance during the test. For instance, Huang [1996] describes an adaptive testing algorithm, CBAT-2, which generates content-balanced questions and SIETTE [Guzmán and Conejo, 2002] is an example of system underpinned by this technique. CAT relies on the Item Response Theory [Hambleton et al., 1991], according to which a student who is answering questions correctly will be gradually administered more difficult questions and vice versa. The student's knowledge level estimation depends on the number of questions answered correctly and, on the difficulty level of the answered questions. Currently, no existing free-text CAA system has included adaptability capabilities yet.

4.1 Techniques

Mitchell et al. [2002] classified the techniques for automatic scoring of free-text responses in three main kinds, *statistical*, *Information Extraction* and *full natural-language processing*. The statistical approach, when it is only based on keyword analysis, has usually been considered a poor method, given that it is difficult to tackle problems such as synonymy or polysemy in the student answers, it does not take into account the order of the words and it cannot deal with lexical variability. On the other hand, a full text parsing and semantic analysis is hard to accomplish, and very difficult to port across languages.

Information Extraction is in the middle of the statistical and the full natural-language processing approaches. It only requires shallow NLP without doing an in-depth analysis and it is more robust than just keyword analysis. However, it still needs training and some lexical resources such as corpus to do the training. To these three main categories, I have added other three: clustering, comparison of semantic networks and hybrid approaches, because they are present in the current free-text CAA systems. It is also important to notice that there may be other techniques which are not considered here as the systems that implement them have

become commercially available, and thus, their implementation details are not longer published in scientific forums.

Therefore, I propose the following classification of free-text scoring techniques:

- **Statistical techniques**: In general, all systems that rely on a statistical analysis of one or several features of the texts should be considered in this category. They usually need an initial training phase to calculate the parameters of the system. They do not use complex NLP techniques and, in most cases, the texts are only processed with a tokenizer and a sentence splitter. As a consequence, they should be easy to port across languages and domains. In particular, there are several subcategories:
  - **Simple keyword analysis**: It is the simplest technique and consists in looking for coincident keywords or n-grams between the student text and the teacher model. This method cannot extract a representation of the meaning of the student answer nor deal with synonyms and polysemous terms. Consequently none of the systems studied is based solely on this technique. On the other hand, they can be included as an auxiliary module as, for example, the Vector Space Model (VSM) [Salton et al., 1975] implementation used in E-rater. In VSM, texts are represented as vectors, and dimensions correspond to words, so that documents can be compared by calculating the cosine of the angle of their associated vectors. A higher cosine angle means a higher similarity with the reference text and, therefore, a higher score.
  - **Surface linguistics features analysis**: In this subcategory, I include systems that require (a) a list of features that are going to be measured, (b) a training phase to discover the weights for each one of them and (c) a calibration phase to adjust the weights to the optimal values. For evaluating an essay, (a) the processing of the essay by looking for these features and (b) using them as the independent variables in the linear regression function whose result is the score. For instance, PEG [Page, 1994] is based on this analysis, although it also uses some additional software such as a grammar checker, a program to identify words and sentences, an electronic dictionary, a part-of-speech tagger and a parser. The main drawback of this technique is to choose which features are going to be considered.
  - **Latent Semantic Analysis**: It is a complex statistical technique that was initially developed for indexing documents and Information Retrieval [Deerwester et al., 1990]. Nevertheless, it can also be applied to automated essay grading [Haley et al., 2003]. In this field, this technique serves to extract the conceptual similarity between the student's candidate text and the teacher's reference text by looking for repeated patterns between them. According to Dessus et al. [2000] this approach is quite robust and proves its name by finding the hidden relationships between words that could be in different documents or between documents that do not share words. The reason for this fact is that what causes two words to have similar meanings is that they change the meaning of passages in which they occur [Landauer et al., 1998].
    LSA might be described in the following way [Whittingdon and Hunt, 1999]:
    1. `The training phase`: In this step, it is necessary to calculate the weights for the

vectors that represent the reference texts. It has the advantage of not requiring a manual marking. Moreover, they could be as big as a textbook or as small as a short paper.

2. `The test phase`: Each student text has to be represented in LSA. It implies several transformations:
   (a) *The matrix representation*: First of all, certain stopwords are removed and words are stemmed, so that there are less words and all of them will be different and meaningful. LSA is a procedure to reduce the dimensionality of the space, i.e. to reduce the size of the matrix, where each row represents a word and each cell its frequency for each context (e.g. a sentence or a paragraph). Therefore, the number of dimensions will be as large as the size of the vocabulary in the language. This matrix can be considered as the VSM representation of the contexts (sentences, paragraphs, or documents) studied.
   (b) *The tuning of the matrix weights*: Now the relevance of each word in the passage is measured. This is done by looking at the frequency of the word in all the contexts and if it is high for every one, this word would not be very indicative of just one of them. This idea is very similar to the used by the inverse document frequency (IDF) weighting [van Rijsbergen, 1979].
   (c) *Singular Value Decomposition (SVD)*: The original matrix is decomposed into the product of three orthogonal matrices. One of them is diagonal and its values are the singular values (the eigenvalues) of the original matrix.
   (d) *Dimensionality reduction transformation*: In order to find the relations between words and contexts, it is necessary to reduce the rank of the diagonal matrix. It is in this transformation where the hidden relationships are detected.
   (e) *The reconstruction*: This new diagonal matrix is multiplied by the other two matrices outputs of the SVD and the result is the weighted version of the original matrix that is the LSA representation of the text.
3. `The result phase`: Once we have the LSA representation of the student text is compared against the LSA model representations and their similarity is computed.

IEA, Apex Assessor and RMT are underpinned by LSA. Besides, this technique has also been employed in other fields such as human tutors to evaluate the quality of the student's answers [Wiemer-Hastings et al., 1998]. Some modifications have also been tried, as Kakkonen et al. [2006] used Probabilistic Latent Semantic Analysis (PLSA) [Hofmann, 2001]. PLSA adds a sounder probabilistic model to LSA based on a mixture decomposition derived from the latent class model. This results in a more principled approach which has a solid foundation in statistics. However, it also has overfitting problems. In fact, the results achieved with PLSA are quite similar of those achieved with LSA.

  - **Text Categorization Techniques**: In this category systems that face the automatic essay grading as a classification problem are considered. The common practice is having an initial training phase to give weights to the relevant features for the final score and then using the model for the classification. An example is the use of Bayesian networks to classify the document as good or bad [Larkey, 1998] or having several score categories and assigning the student text one of them and associated to it his or her score. Larkey's system and BETSY are based on these techniques. In particular, BETSY has the possibility of choosing the bayesian network model to employ: a Multivariate Bernouilli Model (MBM) or a Bernouilli Model (BM) [Rudner and Liang, 2002]. According to Little [2001] MBM should be used with small vocabulary sizes and BM with larger ones. However, none of these models is complete, because the first one only takes into account the fact that an attribute is in the text or not, ignoring the frequency of use; and the second model does not consider the word order. Therefore, these systems are usually used with additional modules, such as one to find the similarity between essays using the K-Nearest Neighbor technique with a retrieval system like Inquery [Callan et al., 1995], another module to stem the words [Porter, 1980], another one to remove the stopwords [Mitchell, 1997] or even another one that implements neural networks techniques. A more advanced Bayesian model is the three-level hierarchical model called Latent Dirichlet Allocation (LDA) [Blei et al., 2003]. It has been tested in AEA although with worst results than using LSA.

- **Full natural-language processing**: NLP is the application of computational methods to process natural language. Burstein et al. [2001] cited tools such as syntactic parsers to find the linguistics structure of a text [Abney, 1996] and rhetorical parsers to find the discourse structure of a text [Marcu, 2000]. Besides, Williams and Dreher [2004] employed electronic thesaurus to extract lexical information and a specifically designed chunking algorithm to extract noun phrases and verb clauses. C-rater and PS-ME are also underpinned by these techniques. Their combination improves the use of statistics by involving a deep text parsing and a semantic analysis in order to gather more information to effectively assess the student's answer. On the other hand, it is hard to accomplish and more difficult to port across languages.
- **Information Extraction (IE) techniques**: Information Extraction consists in acquiring structured information from free text, e.g. identifying Named Entities in the text and filling in a template [MUC7, 1998]. It can be considered as a shallow NLP technique, as it usually does not require an in-depth analysis of the texts. IE may be used to extract dependencies between concepts. Firstly, the text is broken into concepts and their relationships. Then, the dependencies found are compared against the human experts to give the student's score. For example, Automark and ATM are based on this approach. An example of mark scheme is Figure 4.2.

  Pattern-matching is a technique commonly used for IE. It consists in looking for specific information in the student's answer in order to fill in the template that the human experts

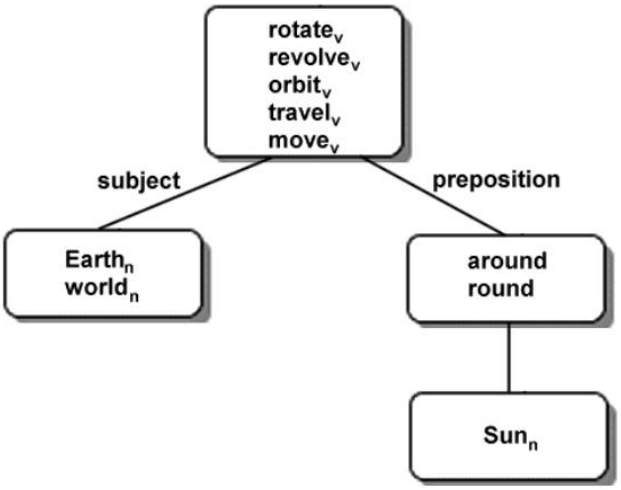


Figure 4.2: Example of a scheme used in Automark to score the answer to the question like "*What movement relates the Earth and the Sun?*" (Mitchell et al., 2003).

have previously done. The filled template is compared against the model to calculate the final score. For instance, SEAR is underpinned by this technique.

- **Clustering**: The systems in this category group essays that have similar words patterns to form a cluster with the same score. For example, Indextron [Mikhailov, 1998] is a clustering algorithm that is being used in the IEMS system.
- **Comparison of semantic networks**: A recent technique applied to automatically scoring essays is the one provided by Lutticke [2005]. It consists in comparing semantic networks expressing the answer of the student with the model semantic network given by the instructors. The comparison is currently done by focusing on which nodes appear in each net and which edges relate them. It is implemented in the MRW system.
- **Hybrid approaches**: It is also possible to take advantage of the better features of several techniques in order to improve a system. For instance, E-rater makes use of VSM for capturing the use of vocabulary and performing the topical analysis, and the rest of the phases are based on NLP; Auto-marking relies on NLP and pattern-matching, and CarmelTC on machine learning techniques and a bayesian neural network classification.

4.2 Evaluation procedures

It would be pleasing to have a standard test set and evaluation metrics to evaluate systems for CAA of free text answers in order to allow a reliable comparison of all these systems and to avoid the problem exposed by Whittingdon and Hunt [1999] who warned that, before admiring the performance of a system, a reflection should be done about the metrics used by the authors. For instance, they said that ETS I results could be overvalued since it only scores answers as correct or incorrect and thus, the agreement between the teacher and the system is easier to achieve. However, given that there is not a standard test set or metric, this section describes

Wished Function	Percentage of respondents
Indicate errors	57.5%
Mark syntax	47.5 %
Provide error statistics	47.5%
Mark non-native speakers writing	42.5%
Produce letter grade	42.5%
Mark organization of ideas	40.0%
Mark surface features	37.5%
Mark rhetorical structure	37.5%
Mark topic content e.g. look at vocabulary	35.0%
Give individual feedback	35.0%
Mark holistically	35.0%
Mark knowledge content e.g. look at semantics	32.5%
Mark analytically	32.5%
Mark according to disciplines	30.0%

Table 4.2: Lecturers' expectations about what a CAA of free text answers software system should provide them in order to be useful (Darus et al., 2001).

Areas of feedback	Percentage of respondents
Errors in the essay	84.2%
Organization of ideas	65.3%
Coherence of text	63.2%
Rhetorical structure	60.0%
Their English dominion	53.7%
Knowledge content	52.6%
Topic content	51.6%
Creativity	51.6%
Style of writing	50.5%
Syntax	40.0%

Table 4.3: Students' expectations about what a CAA of free text answers software system should provide them in order to be useful (Darus et al., 2001).

some expected requisites from teachers and students and the evaluation metrics most commonly used.

4.2.1 Requisites

Darus and Stapa [2001] were concerned with the teachers' requisites and Darus et al. [2001] with the students' requisites in order to trust a CAA of free text answers system. Therefore, they surveyed twenty-two Economics and Business lecturers, twenty-six Arts and Social Science lectures and forty Language and Education lecturers about what they ask to a CAA of free text answers software system in order to consider it a valid and reliable system that they would use with their students. Their results are shown in Table 4.2.

Darus et al. [2001] also surveyed 190 students from the Faculty of Language Studies in order to find out which is the most important factor for them in a CAA system and the answer was unanimous: the feedback. The dispersion was found in the relevance of the different feedback areas as shown in Table 4.3.

4.2.2 Metrics

- **Pearson correlation or inter-rater reliability**: It measures the standard correlation, that is, how much the teachers' scores or true scores (X) are related with the systems' scores (Y). It is calculated by applying Equation 4.1. It is suitable whenever answers are evaluated with a numerical score. Sometimes the true scores are the result of the average consensus of several teachers.

$$Correlation(X,Y) = \frac{covariance(X,Y)}{standardDev(X) \times standardDev(Y)} \tag{4.1}$$

- **Spearman or nonparametric correlation**: It is based on ranking the two variables (teachers' and students' scores) to discover the strength of the link between them. That is, the Spearman rank correlation (r_s) is just the Pearson correlation between the ranks. Hence, it will never be negative and the process is done with rankings from the numerical score. The formula to calculate r_s is Equation 4.2, where 6 is a constant, n the number of paired rankings and $\sum d^2$ the difference in ranks on the two variables, squared and summed. As above, the true score could be the resulting average value of some teachers' scores consensus.

$$r_s = 1 - \frac{6 \times \sum d^2}{n \times (n^2 - 1)} \tag{4.2}$$

- **Exact agreement**: It measures the percentage of times that the system and the human rater have scored just the same value. Although it is a fact that the scores given by humans depend on the human, counting how many times the system scores the same value than the human is considered quite illustrative of the system performance [Larkey, 1998]. As above, it is useful when the answers are evaluated with numerical scores that can be classified into several categories.
- **Adjacent agreement**: It measures the percentage of times that the system and the teacher only differ within one point [Larkey, 1998]. It could be further extended to the Equation 4.3, where Θ is the accepted threshold between the teachers' scores and the system's ones. It is useful whenever the answers are evaluated with numerical scores.

$$AdjacentAgreement = \%(|Truescore - Systemscore| < \Theta) \tag{4.3}$$

- **Mean and Standard Deviation**: It takes into account the dispersion around the mean values of the teachers' and the system scores [Vantage, 2000]. It is one of the less used metrics as it is less informative. For instance, the mean and standard deviation of the teachers' and system's scores could be the same and still the system had been completely wrong giving an inverse set of scores.
- **Kappa measure**: Kappa is a measure of agreement, that is currently gaining popularity as a measure of a scorer reliability by comparing the observed levels of agreement with the levels of agreement expected by chance. The formula for kappa (k) is Equation 4.4, where

Oa is the observed count of agreement, Ea is the expected count of agreement, and N is the total number of respondent pairs [Cohen, 1960, Kraemer, 1982]. It is useful when the answers are evaluated with numerical scores that can be classified into several categories.

$$k = \frac{Oa - Ea}{N - Ea} \tag{4.4}$$

- **F-score**: It combines the precision (p) and recall (r) values according to a certain parameter β.

$$p = \frac{\texttt{number of correct elements found}}{\texttt{number of elements found}} \tag{4.5}$$

$$r = \frac{\texttt{number of correct elements found}}{\texttt{total number of elements}} \tag{4.6}$$

$$\texttt{F-score} = \frac{(\beta^2 + 1) \times p \times r}{\beta^2 \times p + r} \tag{4.7}$$

Finally, another indicator of the goodness of a free-text CAA system can be based on the false alarm rate, which measures the percentage of times that the system could not evaluate a student essay because it was too different from the human models the system had been provided [Rosé et al., 2003].

4.3 Existing free-text CAA systems

In this section, the current free-text CAA systems are presented in alphabetical order to study their main features. For each of them, a brief introduction is presented, followed by a short system description and evaluation.

4.3.1 AEA

The Automatic Essay Assessor (AEA) [Kakkonen et al., 2005] was created in the Computer Science Department of the University of Joensuu in Finland. It is able to assess essays written in Finnish by comparing the student's essay with a set of assignment-specific texts corpus such as textbook passages, lecture notes, etc.

First of all, as Finnish is a morphologically complex language, they have to process the text with a morphological analyzer constraint grammar parser for lemmatization and a syntactic parser. Next, they originally applied the Latent Semantic Analysis (LSA) technique to the reference corpus. However, they have recently also tried Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA) in this phase. In any case, what they obtain is the representation to which compare the human graded essays and determine the threshold similarity values for each grade category. Finally, the LSA, PLSA or LDA representation of the student's essay is compared to the LSA, PLSA or LDA representation obtained in the previous phase and the similarity value of the essay is matched to the grade categories according to their limits to determine the correct grade.

To evaluate the performance of the system, they carried out an experiment using three essay sets collected from courses on education, marketing and software engineering summing a total of 100-150 essays. They tested all the possible dimensions for LSA (i.e. from two to the number of passages in the comparison materials) and the same number of dimensions for PLSA (just to make fair the comparison since PLSA has no restrictions in the number of latent variables). The results indicated that the best technique between LSA, PLSA and LDA is: LSA achieving 75% correlation between the automatic grades and the human grades for the same set of questions.

4.3.2 Apex Assessor

Apex Assessor [Dessus et al., 2000] is integrated inside the Apex web-based learning environment. When students want to study a topic in Apex, they only have to select it and start reading. A final review is done to assess the students' progress in which open ended questions are asked. The Apex Assessor is the responsible of selecting these questions and evaluating them.

Apex Assessor was created in the year 2000 by Dessus, Lemaire and Vernier in the Laboratorie des Sciences de L'Éducation in the Université Pierre-Mendés in France. According to its authors, the aim of the assessment process guided by Apex Assessor is not just summative but formative. Dessus et al. want to engage the students in an iterative improvement process in which they write their texts, then they receive feedback about the outline and the coherence of their essays in order to give the students the possibility of rewriting their essays and send them again.

This system is underpinned by LSA. Thus, it needs a set of unmarked texts for training. This set includes non-technical French texts too to allow the system to deal with non domain terms that might appear in the student answer.

Apex Assessor has three main modules:

- **Content-based assessment module**: It compares the student LSA representation answer with the LSA model.
- **Outline assessment module**: For each paragraph in the student's text, the most similar portion of the course is showed so that the student can be given an outline view of the essay.
- **Coherence assessment module**: It measures the semantic distance between sentences with LSA. Hence, if the proximity between two consecutive sentences is below a threshold a coherence break is detected and the student is warned.

To analyze the system performance, Dessus et al. took 31 essays of a graduate course on sociology of education and typed them into the computer in order to compare the teachers' grade with the Apex one. The result was 59% correlation with $p < 0.001$.

Finally, it is important to mention that one problem Dessus et al. found out was that very short students' answers could achieve high scores. To solve it they have set up the system in order to be stricter with texts that do not have at least 300 words.

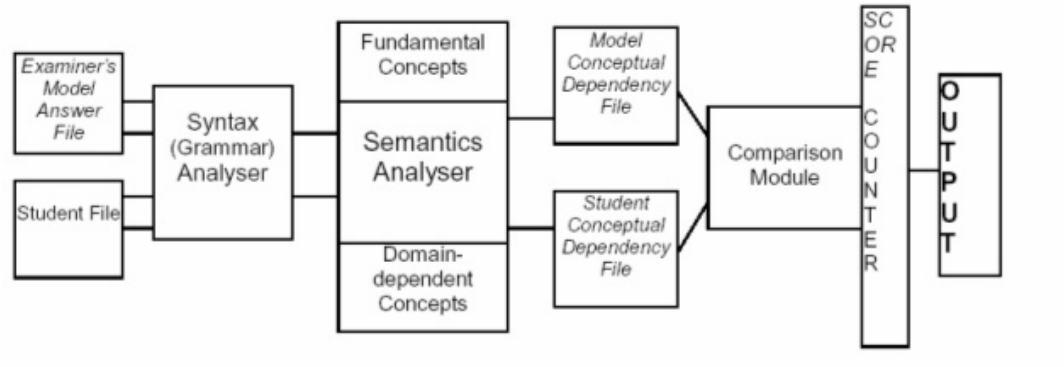


Figure 4.3: Architecture of the ATM system (Callear et al., 2001).

4.3.3 ATM

The Automated Text Marker (ATM) [Callear et al., 2001] was created in the year 2001 by Callear, Jerrams-Smith and Soh in the Portsmouth University in the UK. They were so convinced that both content and style should be taken into account that they designed their system in order to give two independent scores, one for each aspect and to leave the teacher the task of combining them to give the final grade.

ATM relies on IE techniques to assess students' essays. The system architecture is shown in Figure 4.3. It is important to highlight the syntax and semantics analyzer:

- **The syntax analyzer**: It checks the grammar of each input sentence. According to Callear et al. this can be done successfully. A codification in Prolog is given by Callear et al. [2001].
- **The semantics analyzer**: The system looks for concepts in the text and their dependencies, then a pattern-matching Prolog procedure is performed between the dependency groups from the student's answer and the reference model.

According to its authors, ATM works better assessing short answers to factual questions (e.g. in Prolog programming, psychology and biology-related fields). To my knowledge, Callear et al. have not yet published information about their system's performance.

4.3.4 Automark

Automark [Mitchell et al., 2002] [http7] was created in the year 1999 by Mitchell, Russell, Broomhead and Aldridge from the University of Liverpool and Brunel University in UK. At the beginning, it was an academic work but in the year 2002 they founded their company the so-called Intelligent Assessment Technologies and they started using it commercially. Incidentally, in the year 2002 it was made available in ExamOnline [http8] just for registered users.

The aim of the system is mostly summative, that is, to grade the style and the content of a student essay in order to say whether it is acceptable or not according to the criteria specified by the teacher to the system.

AutoMark uses IE techniques [Cowie and Lehnert, 1996] and some NLP techniques to ignore some mistakes in spelling, typing, syntax or semantics that should not be taken into account. The AutoMark assessing process is the following: in the first step, human experts develop some computerized reference mark schemes for the acceptable and unacceptable answers; in the second step, the system standardizes the punctuation and spelling of the student's text; the third step applies the Sleator and Temperley [1991] parser to identify the main syntactic constituents in the student's text and their relationships; the fourth step applies the pattern-matching module that will look for the scheme templates features in the syntactic constituents of the student's answer, trying to cover multiple paraphrasing; and finally, the fifth step processes the output of the pattern-matching module and generates the feedback for the student. It usually consists only of the score, but it might be possible to produce additional information.

The system has been used in the Brunel University to test Java knowledge of first year engineering students, and it has also been applied to assess answers from the 1999 statutory national curriculum assessment of science. In this case, students were 11-year-old pupils, and there were four types of questions: single word generation, single value generation, generation of a short explanatory sentence and description of a pattern in data. The correlation achieved ranged between 93% and 96%.

Finally, four problems can be identified: to correctly identify misspelled words, to correctly analyze the sentence structure, to identify an incorrect answer, and to assess information that is not represented in the mark scheme template.

4.3.5 Auto-marking

Auto-marking [Sukkarieh et al., 2003] [http9] was developed by Pulman, Sukkarieh and Raikes in Oxford and in the Interactive Technologies in Assessment and Learning (ITAL) Unit of the University of Cambridge Local Examinations Syndicate (UCLES). Its aim is not to automatically score high-stakes exams, but to help in low-stakes ones. Each exercise is given a value between 0 and 2, where 0 means incorrect, 1 partially correct or incomplete, and 2 correct and complete.

This system relies on a combination of NLP and pattern-matching techniques. It consists of three modules:

- **Customization and shallow processing module**: Firstly, it uses a Hidden Markov Model part-of-speech tagger, and two finite-state machine chunkers to chunk the noun and verb phrases. Sometimes, an additional manual tuning is necessary.
- **The pattern-matcher module**: It is very similar to the one used in Automark, that is, human experts have to design the information extraction patterns and then the students' answers are compared against them. Appelt and Israel [1999] emphasized the importance of designing good rules. Moreover, Pulman et al. devised a language to express the rules for finding the Information Extraction patterns automatically from the human orders.
- **The marking algorithm module**: These rules are organized in classes and the algorithm described in Sukkarieh et al. [2003] matches them with the student's processed

answer to score it.

The system has been applied with answers from the GCSE exam of Biology with 88% of exact agreement between the teacher and the system. On the other hand, the authors claimed that this system is not suitable for subjective general opinions and therefore it should not be used in that areas.

The main problem they encountered was the inaccuracy of taggers, which do not have enough knowledge about Biology. Besides, they stated that their system cannot deal with students' inferences and with contradictory or inconsistent information.

4.3.6 BETSY

The Bayesian Essay Test Scoring sYstem (BETSY) system [Rudner and Liang, 2002] was developed between 2001 and 2003 by Rudner and Liang at the College Park of the University of Maryland with funds from the U.S. Department of Education. According to Rudner and Liang [2002], its aim is to classify essays using a four point nominal scale (e.g. extensive, essential, partial, unsatisfactory) taking into account both the content and the style.

BETSY is underpinned by naive bayesian networks. The user is given the possibility of choosing one of two models: Multivariate Bernouilli Model (MBM) and Bernouilli Model (BM). Rudner and Liang claim that BM is quicker as it only looks if certain features are present while MBM takes into account the uses in which these features have been employed. A comparison between both models is done in McCallum and Nigam [1998] and, they suggest that MBM with a large vocabulary is more accurate than BM. Although, as Rudner and Liang warn, it might be different with students' essays.

BETSY has the possibility of stemming the text and removing the stopwords, this might improve the text classification task [Mitchell, 1997]. Besides, as it is very CPU demanding, so the authors thought of adding the possibility of purging infrequent words and phrases that appear less than 5 times per thousand.

The system has been used to assess Biology items for the Maryland High School and the results were that BM achieved 80% accuracy and MBM 74%. Furthermore, Rudner and Liang say that their system could be applied to any text classification task.

4.3.7 C-rater and E-rater

C-rater [Burstein et al., 2001] [http10] and E-rater [Burstein et al., 1998] [http10] have been developed by the American Educational Testing Service (ETS) organization. The main goal of C-rater is to distinguish if the student answer is right or wrong in conformity with its content, while the main goal of E-rater is to produce a holistic score based on the organization, sentence structure, and content of the essay. E-rater is included in the web-based system Criterion and C-rater in Alchemist.

The process of assessing a student answer with E-rater requires first of all, to train the system with at least more than two hundred texts about the topic to assess. Next, a feature analysis is performed in which a parser breaks the text in syntactic structures. The discourse

module looks for the terms employed and the rhetorical structures. Finally, the topic module takes into account the vocabulary that is being employed using Vector Space Model (VSM). This module can be applied to the whole essay or to each of the segments discovered by the discourse model. The model is next built by running a forward-entry stepwise regression with the output of the feature analysis to finally give the student's answer score. Whenever E-rater is not able to score the text because it is too short, or too different from the rest, it generates an advisory message [Burstein et al., 2001].

Since 1999, E-rater has scored over 750,000 GMAT essays with an agreement about 97% with the other grader. Burstein and Chodorow [1999] stated that the system can also be used to evaluate non-native speakers' writing.

C-rater is very similar to E-rater. In fact, their main differences are that E-rater focuses on the style, while C-rater on the content; that E-rater assigns a holistic score, while C-rater only identifies whether the response contains specific information necessary to be correct; that E-rater is partly based on the rhetorical structure of an essay, while C-rater is more based on a predicate-argument structure; and that E-rater needs a larger training set.

C-rater has been usually applied to formative low-stakes tasks, as for example the review short questions at the end of each chapter in a textbook. When C-rater was used in a small-scale study with a university virtual learning program it achieved over 80% agreement with the instructor and, according to Leacock [2004], when it was used in a large-scale assessment to score 170000 short-answer responses to 19 reading comprehension and five algebra questions, the result was 85% accuracy.

4.3.8 CarmelTC

Carmel is a Virtual Learning Environment system that has been recently incorporated a new free text assessment module called CarmelTC [Rosé et al., 2003]. This module has been developed at the University of Pittsburgh by Rosé, Roque, Bhembe and Vanlehn. CarmelTC has also been used in the tutorial dialogue system Why2 [VanLehn et al., 2002]. Apart from giving the student a score, it can be used to find out which set of correct features are present in student essays.

CarmelTC relies on the combination of machine learning classification methods using the features extracted from the Carmel's linguistic analysis of the text and the Rainbow Naive Bayes classification [McCallum and Nigam, 1998] .

The procedure for assessing a student's answer is the following: the first step is to break the text in sentences, the second step is to use the bayesian network to look for the possible correct feature that represents each sentence, in order to generate a vector indicating the presence or absence of each correct feature and finally, the third step induces the rules for identifying sentence classes based on these feature vectors with the ID3 tree learning algorithm [Quinlan, 1993].

It can be applied to many several domains, including causal ones such as physics, that are out of the limits of traditional bags of words approaches. In CarmelTC, thanks to the functional

relations found by Carmel, they can be successfully processed.

The system was tested with 126 physics essays, and the results were 90% precision, 80% recall and 8% false alarm rate.

4.3.9 EGAL

The Essay Grading and Analysis Logic (EGAL) [Datar et al., 2004] [http11] is a system developed by a group of American students. It is a source open system based on four criteria: gibberish detection, relevance to the question, identification of facts and their accuracy. They can be used as independent modules or together. In fact, according to its authors, it is more efficient together as whenever a sentence is marked as gibberish, there is no point in continuing studying if it is relevant, as well as if the sentence is marked as irrelevant as it is not necessary to continue checking whether it is a statement of fact.

Gibberish can be semantic or syntactic. In order to determine whether a certain sentence is semantically gibberish or not, the stop words (without lexical meaning) are removed and the rest of the words are stemmed. Thus, it can be calculated its semantic similarity using WordNet [http12]. The mean of the semantic similarities values is the semantic coherence (s) and the percentage gibberishness value is calculated as $100 \times (1 - s)$. Whenever it is above a certain threshold, the sentence is flagged as semantic gibberish. Provided that the sentence is not semantic gibberish, it is checked whether it is syntactic gibberish by parsing the sentence with the module Lingua::LinkParser and looking at the percentage of unknown words and unused links in the sentence. Whenever it is above a certain threshold, the sentence is flagged as syntactic gibberish. The relevance is calculated by calculating the similarity using WordNet between the words of the essay and words in texts about the topic under assessment. Finally, the statement of facts are identified by looking at some rules such as that the sentence is not in future tense and that it contains fact words instead of opinion words.

The system has not been evaluated in depth yet. The authors only report results over seven essays. Thus, the sample is too small to infer any general conclusion. Nevertheless, they state that so far the system's performance has been satisfactory and, that the detection of gibberishness, relevancies and statement of facts is being as expected.

4.3.10 IEA

The Intelligent Essay Assessor (IEA) [Foltz et al., 1999] [http13] was created in the 1997 year by Landauer, Foltz and Laham. It was originally conceived as an academic product but, some years later, they founded their own company called Knowledge Analysis Technology. They are now in the process of patenting their system. Moreover, IEA cannot be executed in an ordinary PC but on secure web servers placed in their company in USA. The authors claimed that as IEA is a web-based application it only takes 20 seconds for students to receive their feedback.

The main goal of IEA is to assess the knowledge conveyed in the essay, rather than its style, syntax or argument structure [Foltz et al., 1999, Landauer et al., 2001].

IEA is underpinned by LSA. This statistical technique has been briefly exposed in section 4.1.

More information about LSA can be found in Deerwester et al. [1990], Landauer et al. [1997], Landauer and Dumais [1997]. One of its main advantages is its language independence, with the restriction that it is not able to process too complex morphological structure of the language.

According to Chung and O'Neill [1997] three main modules can be distinguished:

- **The content module**: It is the most important module. It uses the LSA vector to extract the quality score as the weighted average of the scores for the k most similar calibration essays and the domain relevance score as the length of the essay's vector.
- **The mechanics module**: Punctuation and spelling are analyzed in order to grade the essay's mechanics.
- **The style module**: It takes into account the essay's coherence, which is measured with the LSA value of relatedness among contexts, and the essay's grammar, which is measured with the LSA value of resemblance between the grammatical structure of the essay's sentences and the sentences of the model.

It is also possible to perform synonym recognition in order to treat several synonyms with similar meanings as the same word. The system is also able to identify if students have based their essays more in one reference text or in other. Therefore, it can give the score and feedback to the students with the subtopics that are not enough covered in their essays and links to the reference texts. The system will allow them to resend their essay with the suggested modifications to improve it.

Another technique employed is the anomalous essay checking, that is, the use of a flag to warn the teacher that the essay is too different from the others to reliably assess it and that s/he should review it because maybe the student is having difficulties or maybe s/he is trying to cheat.

According to its authors, IEA can be used in many different applications within education, from the simple consistency checker, to help teachers to discover cheating and plagiarism, to the formative and summative assessment of the essays. It requires an initial training but it is not human supervised. The only input is a set of texts about the topic to evaluate.

IEA has been tested in the military environment with $\sim$ 2000-word essays achieving 0.35 inter-reliability between the teacher and the system (between teachers it was lower, 0.31) [Streeter et al., 2003]. IEA has also been used for psychology, medicine and history texts, achieving 80%-90% exact agreement when a 0 to 100 scoring scale was being used.

Landauer et al. stated that one problem their system has is that it does not take into account the word order. Thus, it cannot interpret sentences in which word order is the discriminant factor. Besides, it is easily tricked because it does not perform any syntactical or grammatical analysis.

4.3.11 IEMS

The Intelligent Essay Marking System (IEMS) [Ming et al., 2000] was presented by Ming, Mikhailov and Kuan from the NGEE ANN Polytechnic in Singapore, in 2000. Its aim is both summative and formative.

IEMS is based on the Pattern Indexing Neural Network, the Indextron that performs pattern recognition and in this case the patterns are the words of the texts. Further information can be found in Mikhailov [1998].

This system has been mostly applied to qualitative questions (e.g. biology, psychology, history or anatomy) rather than numerical ones. For instance, taking a 800-word passage entitled "*Crime in Cyberspace*" and asking 85 students of third-year Mechanical Engineering to write a summary of not more than 180 words about the text, IEMS achieved 80% correlation.

4.3.12 IntelliMetric

IntelliMetric [Vantage, 2000] [http14] was created by the company Vantage Learning, after having spent more than ten millions dollars in its development. It is a commercial system whose focus is emulating the human scorer by grading the content, the style, the organization and the conventions of each response using a 1-4 scale [Vantage, 2000]. It has been improving until its current version IntelliMetric 9.3 [Rudner et al., 2006].

IntelliMetric requires an initial training phase with a set of manually scored answers in order to infer the rubric and the human graders' judgments to be applied by the automatic system. From the initial one hundred features that IntelliMetric could take into account, it chooses the most appropriate for the topic under study. For instance, some features could be the focus and unity that indicate the purpose and main idea of the text: the organization and structure, that indicate the logic of discourse; or the conventions, that indicate conformance to English language rules.

Because it is not an academic product, there is little published information about the techniques that it employs. However, Vantage Learning Technologies has stated that IntelliMetric relies on other of their proprietary systems, the so-called CogniSearch and the Quantum Reasoning Technologies. Moreover, they have claimed that they used an Intelligent Artificial approach, because IntelliMetric uses its intelligence to score the students' texts.

This system has extensively been used in schools, high schools and companies. For instance, it has been used with 594 texts written by students aged 11. Using 100 texts for training, they achieved 98% adjacent agreement. Furthermore, it has assessed essays that are not written in English language, such as Hebrew attaining 84% correlation [Vantage, 2001].

4.3.13 Jess

The Japenese Essay Scoring System (Jess) [Ishioka and Kameda, 2004] [http15] is the first automated Japanese essay scorer. It has been created in the National Research Center for the University Entrance Exam in Japan. It examines three features in the essay: rhetoric (i.e. syntactic variety), organization (i.e. how ideas are presented and related in the essay) and content (i.e. how relevant is the information provided and how precise and related to the topic is the vocabulary employed).

For rhetoric assessment, Jess measures a set of items such as the ease of reading, diversity of vocabulary, percentage of big words and passive sentences. For organization, it attempts

to determine the logical structure of the document by detecting the occurrence of certain conjunctive expressions. For content, it uses Latent Semantic Analysis (the training is done using editorials and columns taken from the Mainichi Daily News newspaper as learning models).

Jess has been evaluated with 480 applicants who wrote an essay about the meaning of work in their life. Three experts scored each essay independently. The correlation achieved between the system's and the mean human scores was 57% higher than 48% found between the expert raters' scores. This result was improved in another experiment in which 143 university students were asked to write about "*festivals in Japan*" with 84% correlation between the automatic and by hand scores. Again higher than the interrater correlation that was 73% [Ishioka and Kameda, 2006].

4.3.14 Larkey's system

Larkey had been working on text categorization techniques to assess students' essays in the University of Massachusetts in USA and she produced her system in 1998 for classifying the students' essays as "good" or "bad". It considers both their content and their style [Larkey, 1998, Larkey et al., 2003].

The assessing procedure could be one of the following, or a combination of them:

1. **Bayesian classifiers**: Each document is assigned a probability of belonging to one previously specified category of documents. To achieve this goal, two steps are performed: the first one is the feature selection that removes stopwords, stems the text with the kstem tool [Krovetz, 1993] and looks for the most representative features using Bayesian networks. It next trains using the Lewis binary model, so that 0 means that the feature is not in the text and 1 is just the opposite [Lewis, 1992].
2. **Finding the k most similar reference essays**: The Inquery retrieval system is used to find the k essays closest to the student essay [Callan et al., 1995].
3. **Using eleven text complexity features**: Eleven features are automatically calculated from the text. Some of them are the number of characters in the document, the number of different words in the document, the average sentence length, the average word length and the number of words longer than seven characters [Larkey, 1998].

The score is the result of the linear regression performed with the results of the values for the features, the results of the Bayesian classifiers or a combination of the three methods.

Larkey's system has been applied to essays on social studies, physic questions and legal arguments, achieving 60% exact agreement and 100% one-point-of-difference agreement when all the criteria for assessment were used. Therefore, she has even tried to assess general opinion questions, with the results of 55% exact agreement and 97% for one point of difference. The correlation attained was always above 80% and, in particular, for the general opinion essays, it was 88%.

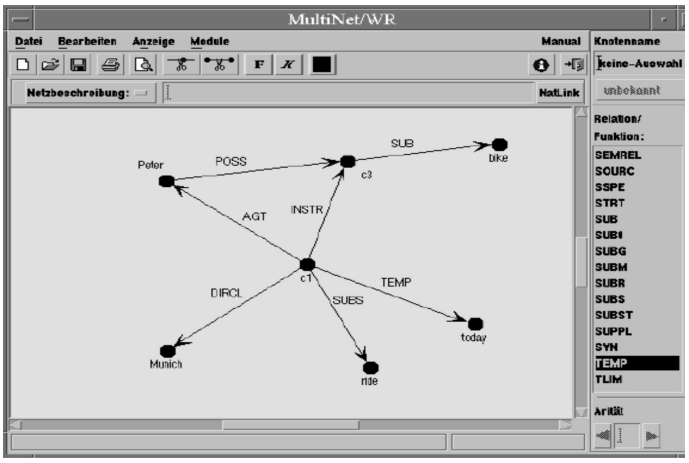


Figure 4.4: A snapshot of the MRW system.

4.3.15 MarkIT

MarkIT [Williams and Dreher, 2004] [http16] is a free-text scoring system that gives feedback to the students about how they have used the concepts in the essays. It has been developed by a research team at the School of Information Systems in the Curtin University of Technology in Australia. It uses propriety technology based on NLP techniques, Latent Semantic Analysis and an electronic thesaurus to process and compare the student's answer with the model answer that has been extracted from a set of e-learning contents. It has been the source of inspiration of the E-Tester system described in Section 3.4.

First of all, it is necessary to train the system by feeding it with 50-200 human graded essays (the assessment is better as more human raters score the test and the score is averaged). In this way, it can be tuned to use multiple linear regression. Next, both the student's answer and the model answer are processed by a specially designed chunking algorithm called Context Free Phrase Structure Grammar parser to identify the noun phrases and verb clauses in them. During this phase, a transformational grammar is used to represent the semantics of the content and the thesaurus to extract lexical information and, the internal knowledge representation of the answers is built. Finally, pattern matching techniques are employed to ascertain the proportion of model answer knowledge present in the student answer that is scored accordingly.

The system has been tested with 390 essays written by year 10 high school students on the topic of "*The school leaving age*". They were asked to write their essays in Microsoft Word document format. Next, these essays were submitted to three different human graders and MarkIT. 200 of them were used as training and the other 190 essays as test. The results for the test were 75% correlation between human scorers and the authors claimed that MarkIT performed as well as human graders (although no numerical information was provided).

4.3.16 MRW

The MultiNet Working Bench (MRW) [Lutticke, 2005] [http17] is a graphical tool to assess student knowledge. It has been created in the Computer Science Department of the FernUni-

versitat in Germany. It is based on the MultiNet paradigm whose core idea is to represent natural language as semantic networks in which the nodes refer to discourse entities and the edges to semantic relations between them. Inner nodes are for more complex concepts and a fixed set of 110 relations has been defined. See Figure 4.4 for an example of semantic network for the sentence "*Today Peter rides his bike to Munich*".

MRW is able to represent, edit and assess semantic networks in MultiNet form. The analysis can be done from the net as drawn by hand by a student or from its natural language reformulation. In any case, the internal representation of the student answer as semantic network and a reference solution is compared using logic inference and the result can be that the text is wrong, with missing fragments, unverified or verified. This result is given to the student as textual and graphical information. For instance, wrong or unverified parts of the students network are marked in red and verified parts in green. The feedback can also be enriched with support hints such as links to literature or examples.

The system is currently subject to further development and extension. Nevertheless, a preliminary version has been used in a practical NLP course imparted in the FernUniversitat with promising results. However, no numerical results are published yet.

4.3.17 PEG

The Project Essay Grader (PEG) [Page, 1966] [http18] was first presented in 1966 by Page in the University of Duke in USA. It focused on the style of the essay.

At the beginning, no NLP technique was used and the system was based only on a statistical approach that consisted in looking for several features (proxes) that represent abstracter ones (trins). According to Chung and O'Neill [1997], PEG considered 28 different proxes such as the title, the average sentence length, the number of paragraphs, the punctuation and the number of prepositions in 1966. Incidentally to score the students' essays, PEG is introduced a number of previously manually marked essays for proxes to calculate the coefficients for the regression equation that finally will give the students' scores. In 1990, the system was improved with a grammar parser and a part-of-speech tagger to improve the proxes discovery. Moreover, in conformity with Shermis et al. [2002], PEG currently includes content, organization, style, mechanics and creativity assessment.

PEG is suitable for most type of essays, achieving 87% correlation between its scores and human ones.

4.3.18 PS-ME

The Paperless School Marking Engine (PS-ME) [Mason and Grove-Stephenson, 2002] [http19] is the system presented by Mason and Grove-Stephenson in the Birmingham University in UK in 2002, and it has also become commercially available. Its assessment objective is both summative and formative, with little or no human intervention. Besides, it can be integrated in a learning management system, or be used as a stand-alone application.

PS-ME relies on NLP techniques to cover Bloom's taxonomy [Bloom, 1956]:

- **Knowledge level**: It exactly corresponds to the Bloom knowledge competence. According to Mason and Grove-Stephenson, it is only necessary to create, from the reference texts, a list of the most relevant concepts that should be present in the student's essay in order to evaluate this level.
- **Understanding level**: It comprises the competencies between knowledge and evaluation in the Bloom's taxonomy: comprehension, application, analysis and synthesis. Mason and Grove-Stephenson refuse to give more details about this level arguing that it is commercially sensitive [Mason and Grove-Stephenson, 2002]
- **Evaluation level**: It matches the Bloom evaluation competence. PS-ME's authors are not too convinced that this level can be effectively measured by a computer, since the teacher usually scores higher a creative opinion of a student than one that is based on a reference text. However, they have included this option that could be based only on the frequency of adjectives and adverbs in the text; or even better, by looking for some syntactic patterns such as "*I think that X...*" or "*It is obvious that X*".

PS-ME requires an initial training phase with at least thirty hand-marked sample texts that could include not only reference texts, but also "negative" texts with a very low score. Besides, Mason and Grove-Stephenson thought that due to processing requirements, their system should not be used for real-time essay grading. Instead, they implemented it as a web-based system that sends the information in XML to a queuing system in the server. Finally, PS-ME does not only give the score but some formative feedback to the student in different areas within the subject.

This system has been applied to low-stakes coursework, National Curriculum Grade and GCSE exam in the academic field. In the commercial field, it has usually been employed by publishers. The main problems found were the difficulty for selecting master texts and the misspellings and bad grammar mistakes that, in words of Mason and Grove-Stephenson, could "throw the system out". To my knowledge, Mason and Grove-Stephenson have not yet published PS-ME results.

4.3.19 RMT

The Research Methods Tutor (RMT) [Wiemer-Hastings et al., 2004] is a web dialog-based tutoring system result of the joint effort of the Computer Science and Psychology departments of the DePaul University in USA. It is the descendant of the related system called AutoTutor [Graesser et al., 2005]. It has been designed to be flexible enough to integrate different tools and techniques for improving tutoring.

As AutoTutor, RMT is based on LSA. This means that first of all it needs to be trained with a set of reference texts. Next, it evaluates the student response by transforming it to its LSA representation and comparing it to the LSA representation of the expected answers. In this way, the intelligent tutor can continue asking the student according to the good or bad answer provided by the student. Currently, they are exploring the possibility of improving the technique by segmenting input sentences into subject, verb, and object parts and comparing

each separately.

The system was integrated as a regular component of the Research Methods in Psychology course in order to find out how students use such systems during a whole term. However, due to technical difficulties with the agent software and some compliance issues with the students, they did not get significant results. Further experiments are planned in the short future [Wiemer-Hastings et al., 2005].

4.3.20 SAGrader

The Semantic Analysis Grader (SAGrader) [Works, 2006] [http20] is the tool offered by the American company Idea Works to assess free-text answer essays. It uses the proprietary QTools developed by the same company to recognize patterns in students' essays and compare them with correct answers. In this way, it can give detailed feedback to students.

According to its authors, the system is not restricted to any domain and its purpose is not to replace the instructor but to assist him or her. In fact, instructors are crucial as they are who introduce the semantic network of knowledge using the SAGrader's developer's module (they only have to do it once per course as it can be reused for later courses). Instructors also have to specify the students' assignment so that the system can decide which elements of the net should be present in the students' answer and, to generate some possible questions to ask the students.

Additionally to the feedback provided to the students, the system can also give feedback to the instructors. In particular, teachers can see, per each student, the date of submission of the essay, the text as has been archived and, concerns or objections with the grade given by SAGrader.

SAGrader has been tested in sociology courses in introductory freshman and sophomore level classes in a university setting. It has been proved its good performance for classes where the primary objective is to assess student's knowledge at an introductory or intermediate level. Moreover, students reported how they like the program because it provides immediate detailed feedback any time of the day or night and gives them an opportunity to revise their paper and improve their grade. In fact, they were able to improve their grades from an average 70% up to 90%. However, no numerical values have been provided to evaluate the accuracy of the marks provided.

4.3.21 SEAR

The Schema Extract Analyse and Report (SEAR) system was presented in the Robert Gordon University in UK to assess both the style and content of the students' essays [Christie, 1999, 2003] [http21]. In general, the system is underpinned by Information Extraction techniques. However, the algorithms for assessing style and content are different:

- **For style**, four steps are needed: to pre-determine the candidate metrics, to have some manually marked system, to calibrate the system in order to find an acceptable agreement between the human expert and SEAR, and to process the student's essay just by looking

System	Domain	Availability
AEA	Marketing and software engineering	Academic
Apex Assessor	Sociology of education	Academic
ATM	Factual disciplines	Academic
Automark	Science	Academic
Auto-marking	Biology	Academic
BETSY	Any text classification task	Free
CarmelTC	Physic	Academic
C-rater	Comprehension and algebra	Academic
EGAL	Opinion and factual texts	Free
E-rater	GMAT exam	Academic
IEA	Psychology and military	Commercial
IEMS	Non-mathematical texts	Academic
IntelliMetric	K-12 and creative writing	Commercial
Jess	General topic essays	Academic
Larkey's system	Social and opinion	Academic
MarkIt	General topic essays	Academic
MRW	NLP course on semantic networks	Academic
PEG	Non-factual disciplines	Academic
PS-ME	NCA or GCSE exam	Commercial
RMT	Research on Psychology	Academic
SAGrader	Sociology courses	Commercial
SEAR	History	Academic

Table 4.4: Domains to which the current existing CAA of free text answers systems have been applied and their availability.

for these features and applying the weight of each metric to compute the score as the result of a weighted linear function.

- **For content**, neither training nor calibrating is necessary. The teacher needs to create some reference schemes. It uses Information Extraction techniques to fill in the students' schemes with the students' data and to compare them against the references.

It has been applied to assess essays about the potted history of Robert Gordon (the founder of the Robert Gordon University). The results attained are from 30% to 59.4% correlation between the system and the human scores.

4.4 Comparison and conclusions

Despite the core idea of all these systems is the same: to compare the student answer with one or more reference texts, a complete objective comparison cannot be done because they use different corpora and evaluation metrics. Nevertheless, in an attempt to put together the different techniques and results provided by their authors, Tables 4.4 and 4.5 are presented with the systems alphabetically ordered.

It can be seen that each system has been applied to a somewhat different assessment area, depending on the technique that they are employing. For example, LSA, that has its focus on the content, is mostly used for the assessment of humanities' essays [Wiemer-Hastings et al., 1998], while IE techniques, that rely on filling schemes, can be used both for factual and non

System	Technique	Evaluation	Language
AEA	LSA, PLSA or LDA	Corr: .75	Finnish
Apex Assessor	LSA	Corr: .59	French
ATM	Pattern matching	–	English
Automark	Information Extraction	Corr: .95	English
Auto-marking	NLP and pattern matching	EAgr: .88	English
BETSY	Bayesian networks	CAcc: .77	English
CarmelTC	ML and bayesian networks	f-S: .85	English
C-rater	NLP	Agr: .83	English
EGAL	NLP and statistics	–	English
E-rater	NLP and VSM	Agr: .97	English
IEA	LSA	Agr: .85	English
IEMS	Pattern matching	Corr: .80	English
IntelliMetric	CogniSearch and Quantum	Agr: .98	English
Jess	Pattern matching	Corr: .71	Japanese
Larkey's system	TCT	EAgr: .55	English
MarkIt	NLP, pattern matching and statistics	Corr: .75	English
MRW	Logical inference	–	German
PEG	Linguistic features	Corr: .87	English
PS-ME	NLP	–	English
RMT	LSA	–	English
SAGrader	QTools	–	English
SEAR	Pattern matching	Corr: .45	English

Table 4.5: Overview of the techniques, evaluation and language of the reviewed free-text CAA systems. Possible metrics are: Corr, correlation; Agr, Agreement; EAgr, Exact Agreement; CAcc, Classification accuracy; f-S, f-Score; and, – for not available. When the authors have presented several values for the results, the mean value has been taken.

factual disciplines. In fact, LSA has performed poorly in causal domains such as research methods [Malatesta et al., 2002]. Moreover, according to Callear et al. [2001] neither Apex Assessor not IEA are suitable to assess short answers where the word order is important.

It is also interesting to mention how free-text CAA has prospered and has not been limited just to English texts and academic purposes. In fact, the level of performance achieved by some of these systems have made possible their use as commercial applications.

There is much discussion and disagreement about which system can be considered as the best. For example, Rudner and Gagne [2001] stated that, among IEA, E-rater and PEG, the best choice for evaluating writing style is PEG. This is because it relies on writing quality features to determine the grades. Besides, it is simpler and it consumes less CPU. On the other hand, IEA and E-rater are better for grading content. However, since IEA can be tricked as it does not perform any NLP processing, E-rater can be thought as the best one at the cost of being the most complex. This opinion is also shared by Williams [2001] who said that in terms of comparison with human markers, E-rater is best, followed by IEA, Appex Assessor, Larkey's system and finally PEG.

All the same, Rudner and Liang [2002] claimed that bayesian networks are the best approach because they are easy to implement, they combine the advantages of PEG, LSA and E-rater and, they are perfectly suitable to assess short essays. Despite that, in conformity with Cucchiarelli

et al. [2000] the main weakness of all these systems is the lack of a very large corpus of essays that may become a reference for everyone interested in automated essay grading.

In conclusion, there is no current system that could be highlighted as the best one by all comparisons. However, it is important to highlight that the technology exists. In fact, just combining the main advantages of each one, the resulting system might be the ideal CAA of free text answers tool.

Part II

Proposal for the generation of students' conceptual models underpinned by free-text Adaptive Computer Assisted Assessment (ACAA)

In this part, it is presented an approach to automatically generate students' conceptual models (i.e. students' network of interrelated concepts) from their answers to free-text Adaptive Computer Assisted Assessment (ACAA) systems. Free-text ACAA systems are the natural evolution of free-text CAA systems in which the assessment is tailored to each student's particular features. In this way, the specific information about each student can be exploited with Natural Language Processing (NLP) techniques to build his or her conceptual model inside his or her student model with the rest of personal data gathered. The procedure has been implemented in the *Will tools* that consist of: Willow, a free-text ACAA system; Willed, an authoring tool; Willoc, a configuration tool to select the NLP techniques to be used; and, COMOV a conceptual model viewer.
This part consists of three chapters (the list of publications that has produced the work described in each chapter is also given):

- Chapter 5 entitled "**Automatic and adaptive free-text assessment for conceptual modeling**" describes the student and domain models as they are automatically generated, as well as the procedure to build them. A step-by-step example is given to illustrate the procedure. The publications related to this Chapter are: Pérez et al. [2005a], Alfonseca et al. [2005], Pérez-Marín et al. [2006a,d,g], Pérez-Marín et al. [2007]

- Chapter 6 entitled "**An example of free-text ACAA system: Willow**" is devoted to Willow, an on-line application with Natural Language Support, able to process free-text answers and automatically score them. Besides, it adapts its behavior to each student and keeps track of his or her use of terms and the relationships between these terms to generate his or her conceptual model. The publications related to this Chapter are: Pérez [2004], Alfonseca et al. [2004a], Pérez-Marín et al. [2006b,c], Pérez et al. [2006f].

- Chapter 7 entitled "**An example of conceptual model viewer: COMOV**" introduces a conceptual model viewer called COMOV, developed to display the generated conceptual models in five different representation formats: concept map, conceptual diagram, table, bar chart and textual summary. Please, notice that the examples given in this chapter are as generated from the data gathered of a real student in an Operating System course. However, given that the student is Spanish, and in order to make the reading of these representations easier to the non-Spanish speaker, the terms have been manually translated into English. Appendix E gathers the source Spanish terms. The publications related to this Chapter are: Pérez-Marín et al. [2006c], Pérez-Marín et al. [2007a].

Chapter 5

Automatic and adaptive free-text assessment for conceptual modeling

On one hand, as it has been seen in Chapters 2 and 3, students' conceptual models are useful for instructors who need feedback to find out how well the concepts exposed in the lessons are being understood by the students. Additionally, students can also benefit from having their conceptual models available, either represented as concept maps or other forms of knowledge representation.

On the other hand, in Chapter 4, the state-of-the-art of free-text CAA systems were reviewed. They could be used as stand-alone applications or integrated into some type of educational system. However, none of them used any kind of student model aimed to tailor the assessment to each student's needs. The most important attempt to provide adaptation capabilities in assessment has been Computer Adaptive Testing (CAT) that adapts the order of the item in a multiple-choice test according to a set of metrics. Notwithstanding, it is just a small subset of the possibilities that adaptive assessment of free-text answers can have. It is because, as some authors have noticed, the adaptation in CATs, depends on the Item Response Theory (IRT) model, which requires a careful item calibration on big items databases [Giouroglou and Economides, 2004].

In this work, it is proposed to combine several techniques from the AH and NLP fields to alleviate the needs previously mentioned: to provide a procedure to fully generate students' conceptual models, to adapt the assessment of free-text answers and, to overcome the limitations of CAT. In order to achieve these goals, it is necessary to evolve to free-text Adaptive CAA systems (ACAA), which incorporate adaptive possibilities to the assessment.

Section 5.1 delimits the scope of the work and makes its underlying assumptions explicit; Section 5.2 gives the definition of the domain model; Section 5.3 gives the definition of the student model; Section 5.4 describes the procedure to generate both the domain and student models using a free-text Adaptive Computer Assisted Assessment (ACAA) system. Additionally, an illustrative step-by-step example is also provided in this Section. Finally, Section 5.5 focuses on the possibility of not only generating one particular student's conceptual model but the whole class conceptual model.

5.1 Scope

First of all, it is important to highlight that there are some kind of systems that are not considered in this work as they are out of its limits. They are: semi-automated computer based essay marking systems, systems that assess the student ability to summarize, systems to improve the student writing skills, Computer Assisted Language Learning systems to assess the ability of writing in not mother-tongue languages and, systems that foster collaborative work along the free-text assessment.

Secondly, this work focuses on the assessment of short free-text answers (i.e. one or two paragraphs). However, it is also important to notice, that some techniques traditionally used for assessment of essays, such as Latent Semantic Analysis, are also applied to the automatic assessment of short answers.

Thirdly, although there exists some free-text CAA systems able to handle hand-written input, this approach is still too challenging and achieves very low results [Allan et al., 2003]. Besides, some students are worried because of their bad calligraphy [Parsons et al., 2003] and thus, they are grateful that they can use the keyboard to enter data. Therefore, in this work, keyboard input is the only input method contemplated.

Fourthly, a constructivist approach for teaching and learning is followed.

Fifthly, the following assumptions have been done in this work:

- Novak's definition of concept provided in Section 2.1 is followed referring to the label of the concepts as "*terms*". Besides, only static relationships among the concepts are considered, because these relationships serve to organize the knowledge according to a certain hierarchy.
- For each area-of-knowledge, instructors are supposed to be able to provide several questions and to structure them into topics. Besides, for each question, they are supposed to be able to write several correct answers (typically 3 or 4, the reference answers or references written in natural language, without templates or any kind of restriction), expressed with different words or grammatical constructions.
- The more similar the student's answer is to the references, the better the answer is and thus, the higher the score and the confidence-value of the terms that appear in the answer is (i.e. students should use the terms in their answers as instructors do in the references).

5.2 Domain Model

In this section, the content and organization of the model of the domain is explained. In this case, the domain is considered as the course or area-of-knowledge under assessment. Furthermore, as the goal is not to teach, but to formatively assess students' answers, fragments of texts related to the domain are not stored. On the other hand, **questions and their correct answers (references) are the core of the model**.

This information is provided by the **instructors** who play a key role in the creation of the domain model. There may be one or more instructors. In particular, it would be convenient

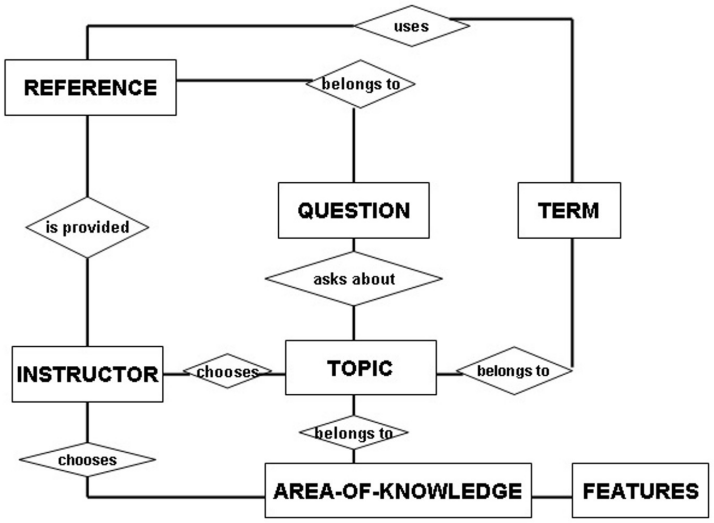


Figure 5.1: Representation of the proposed storage for the domain model as a simplified entity-relation model. Horizontal links should be read from left to right and vertical links from top to bottom.

that there were more than just one instructor as, in this way, the process is less dependent on a particular individual.

Firstly, instructors have to indicate the name of the course to assess, i.e. the name of the **area-of-knowledge**. Furthermore, they have to decide which **features** are going to be considered to adapt the assessment of the course. For instance, if the instructor has chosen to include the feature *Language* with two values: Spanish and English, then all the natural language information stored in the domain model will have to be stored in Spanish and English, and the area-of-knowledge would be "*Operating Systems*" for English and, "*Sistemas Operativos*" for Spanish. Furthermore, if the instructor has chosen to include two features such as *Language* and *Experience* with the values: Spanish and English for Language and, Novice and Expert for the Experience, then all the natural language information stored in the domain model will have to be stored in the Spanish-Novice, Spanish-Expert, English-Novice and English-Expert versions. In this case, the Spanish-Novice version of the area-of-knowledge would be "*Introducción a los sistemas operativos*", the Spanish-Expert version would be "*Sistemas Operativos*", the English-Novice version would be "*Introduction to Operating Systems*" and, the English-Expert version would be "*Operating Systems*".

Secondly, instructors have to provide the names of the lessons of the course, i.e. the name of the **topics** considered. For instance, for the "*Operating Systems*" area (considering that the instructor has not chosen any extra feature for the area-of-knowledge), two topics could be: "*Concurrency*" or "*Scheduling*".

Thirdly, instructors have to provide a set of **questions** per topic. The minimum information that should be given per question is: its statement in natural language; its maximum numerical score; its numerical score to pass the question; its difficulty level in the range low (0), medium (1) or high (2); the topic to which the question is related to and, finally, a set of correct answers

or **references**. It is important to highlight that the references are stored in natural language, just as the instructors write them and, without using any kind of template. For instance, a question in the topic of "*Introduction to Operating Systems*" of the area-of-knowledge "*Operating Systems*" (without considering any extra feature) could be: "*What is an operating system?*" with 0.5 maximum score, 0.3 score to pass the question, medium level of difficulty and the following reference "*An operating system is the application that serves as interface between the user and the processes running by it. A process is a program in execution such as a shell or a database. Examples of operating systems are Windows and Linux.*"

Fourthly, given that students can be promoted, per topic, to be asked more difficult questions as soon as they pass a certain number of questions, or be demoted to easier questions as soon as they fail a certain number of questions, instructors have to introduce which number of questions (usually a percentage of the total number of questions in the topic) has to be passed / failed by the students to be changed of level of difficulty.

Finally, in order to alleviate teachers of some work, **terms** are automatically extracted from the references provided as will be explained in Section 5.4.1. All the same, teachers can later review this list of terms and, modify it by adding or removing terms as they consider more adequate.

Please, notice also here that, as can be seen in Figure 5.1, all the entities (area-of-knowledge, topics, questions, references and terms) stored in the domain model are not stored as independently one from the others. In particular, the area-of-knowledge is stored related to the topics that it comprises, and the name of the topics are related to the questions. Moreover, the terms are associated to the references in which they appear, and the references to the questions to which they are the correct answer.

5.3 Student Model

This section presents a first approach devised to keep a student model in free-text ACAA systems. In this way, it is possible to take into account the students' preferences and personal features to adapt not only the assessment process but also to personalize the appearance of the interface. In fact, the students should have full control over the interface (although default values should also be set for those students who do not want to have their environment personalized). This is important not only from an aesthetic point of view, but also functionally, because it can help students with some disability.

The student model is composed by two components: a first one that is static and introduced by the student, consisting of students' features that do not change, at least during the assessment session such as personal preferences as described in Section 5.3.1 and, a second one that is dynamic, inferred by the system from the behavior of the student. Dynamic information changes according to the information stored in the static component, i.e. depending on the features chosen for the topics to be asked, different questions are asked for these topics. Furthermore, depending on how the students answer these questions, different questions are asked and the students' conceptual models are modified accordingly as described in Section 5.3.2.

Feature	Possible value 1	Possible value 2
Language	Spanish	English
Experience	Novice	Expert
Age	Child	Adult
Session duration	Chronometed	Unlimited
Feedback	Basic	Detailed
Interface	Default	Personalized

Table 5.1: Example of values (two have been shown per feature but it could be a greater number) for a set of features proposed for a free-text ACAA system so that it can offer static adaptation.

5.3.1 Static component of the model

Initially, students should give some contact information such as their name and mail and, choose the area-of-knowledge in which they want to enrol. Please, notice here, that first of all, they will be asked the values for the features considered by the instructors for the area-of-knowledge chosen. Thus, for instance, provided that the instructor has selected the language, experience and age features, students should state their values for each of them. It is important because, regarding:

- **Language**: The statement of the question should be presented in the language of the student. Equally, the answer should be evaluated with the right NLP tools and resources for that language.
- **Experience**: Novice students should be presented simple questions so that they can answer them, whereas more expert students should be presented more complex questions so that they do not get bored with too easy questions.
- **Age**: The statement of the question should be easier for children than for adults. Besides, the references need to be different since children's vocabulary is expected to be simpler.

Additionally, there are some features that are always asked at the beginning of each assessment session, irrespectively of the area-of-knowledge and the student, such as the:

- **Topics**: The student can choose which topic(s) (one or more) s/he wants to review. If none is chosen, then questions from any of them can be asked.
- **Session duration**: The student is given the option of choosing the number of questions s/he wants to answer in one session, as well as specifying the amount of time to dedicate to the session. In this way, the assessment session will last until one of the two conditions is fulfilled, i.e. the number of questions requested is passed or, the time is over.
- **Feedback**: The level of information given to the student. It can range just from the numerical score given to each question answered to the detailed processed answer and teachers' references.
- **Interface**: The appearance of the interface can change according to the students' preferences (e.g. making the text box for the answer bigger, changing the font type, etc.).

All of these features configure what can be called the "*adaptation paths*" that the system should follow for each student. Table 5.1 shows a possible set of features and values that may have been set by an instructor and Figure 5.2 the possible adaptation paths for these values.

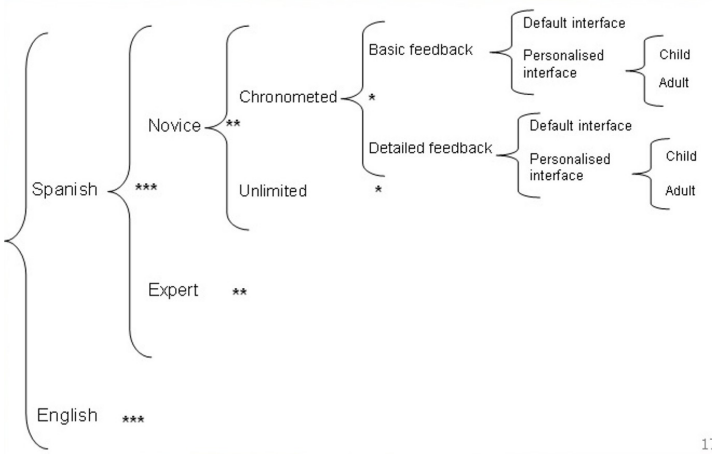


Figure 5.2: An example of adaptation paths from the features gathered in Table 5.1. The whole branching is not shown, instead, asterisks are used to indicate that the same branching should be repeated where they appear (the number of asterisks indicates which branch should be copied).

That way, the free-text ACAA system can present the right statement of the question and assess it using the right versions of the references for each student's profile.

5.3.2 Dynamic component of the model

Static adaptation is quite limited, because once the values are fixed at the beginning of the session, they cannot be changed again during the assessment process. Thus, it is also necessary to record dynamic values such as: the **answer** given by each student to each question with the automatic numerical **score** given by the free-text ACAA system, the **percentage of questions** passed and failed per topic, and the **level of difficulty** that the student is able to pass for each topic (initially, all difficulty-level values are set to low, 0 and from that, they can increase up to medium, 1, or high, 2). In this way, the free-text ACAA system is able to ask questions in the students' level of difficulty so that students do not get bored with too easy questions or they just give up because of too difficult questions. For instance, to the question *"What is an operating system?"* presented in the previous section, an example of correct answer would be: "*It is a program that serves as interface between the user and the hardware. Some examples are Windows and Linux.*".

Additionally, in the dynamic part of the student model, it is also kept the **student's conceptual model**. It can be defined as a **simplified representation of the concepts and relationships among them that each student keeps in his or her mind about an area of knowledge at a certain instant**. It is useful both to be displayed to students and teachers, and to be used by the system to guide the assessment. In particular, teachers can easily see students' strong and weak points, students can get extra feedback, and the system can change the order and content of the questions to focus on the misconceptions or erroneous links detected.

In order to organize the students' conceptual model, a hierarchical structure of knowledge

has been considered, according to which, every concept in the model has the same relevancy. In fact, the higher a concept is in the hierarchy, the more important it is. Three different types of concepts have been distinguished:

- **Basic-concepts (BCs)**: They are what Novak considers a concept. Hence, they can be taken from the terms automatically identified from the references and stored in the domain model. Moreover, given that the goal is to find out the level of assimilation per student of each concept, each of them is associated to a confidence-value (CV) that reflects how well the system estimates that the student knows it. The CV of a concept is always between 0 and 1. A lower value means that the student does not know the concept as s/he does not use it, while a higher value means that the student confidently uses that concept. This CV is automatically updated as the student keeps answering questions according to a set of metrics.
- **Topic-concepts (TCs)**: They are the concepts related to the name of the topics introduced by instructors in the domain model as lessons of the course under assessment. TCs are an intermediate level in the hierarchy as they group several BCs (a BC can belong to one TC or to several TCs but it only appears once in the conceptual model) and, belong to a certain area-of-knowledge. The CV of a TC is calculated as the mean value of the CVs of the BCs that it groups.
- **Area-of-knowledge-concepts (ACs)**: They are the names of the courses. ACs are the highest level concepts as they refer to groups of several TCs. For each conceptual model, only one AC is allowed. The CV of an AC is calculated from the CVs of its related TCs. For instance, for the sample area-of-knowledge given in the previous Section, the AC would be the "*Operating Systems*" course, one TC would be the "*Concurrency*" topic and, one BC would be "*Dekker algorithm*".

Regarding the relationships between these concepts, three types of links have been distinguished according to the type of concepts that they relate:

- **Type 1, between ACs and TCs**: A topic-concept *belongs to* one area-of-knowledge-concept. Type 1 links are extracted from the information of the course provided by the instructors (i.e. which lessons corresponds to each course) and, stored in the domain model. A TC can only belong to one AC.
- **Type 2, between TC and BC**: A basic-concept *belongs* at least to one topic-concept. It can also belong to several topic-concepts. These relationships are important because they give us information about how the basic-concepts are grouped into topic-concepts and, how the students are able to use the BC in the different questions related to each topic. TCs are not linked among them, as the relationships between the topics are already captured by the type 3 links. Type 2 links are extracted from the domain model, in particular, from the relationships between the topics and, the terms found in the references of the available questions to assess this topic.
- **Type 3, between two BCs**: A basic-concept can be *related* to one or more basic-concepts. These links are very important as they reflect how BCs are related in the student's cognitive structure as are extracted from the students' answers. Please, notice

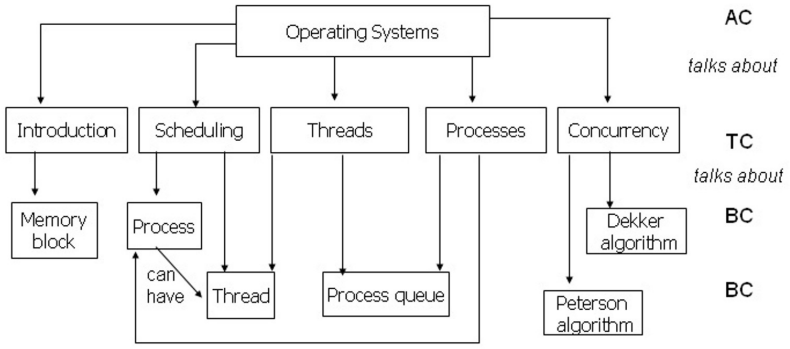


Figure 5.3: Example of the hierarchical structure of the conceptual model.

also that, while type 1 and 2 links are stored both in the domain and conceptual model, type 3 links are only present in the students' conceptual models as they are specific to each student's particular student.

Each link has associated one or more linking words that joins the concepts in the extremes of the link and forms propositions (see Section 2.6). The linking words for type 1 and type 2 links have been fixed as "talks about" (from the higher concept in the hierarchy to a lower concept) or "belongs to" (from the lower concept in the hierarchy to a higher concept). These linking words have been chosen as they serve to structure the knowledge and thus, capture the essence of these type of links. On the other hand, the linking words of type 3 links, as they join basic concepts that appear in the student's answer, are extracted from it. For instance, Figure 5.3 shows a graphical representation of the three type of links considered in the conceptual model with their linking words.

It can be seen that the area-of-knowledge comprises several topics and each topic several concepts. In this way, it can be said that *Operating system* "talks about" *Concurrency* or, the other way around, that *Concurrency* "belongs to" *Operating systems* (example of type 1 link); that *Concurrency* "talks about" the *Dekker algorithm* or the other way around, that the *Dekker algorithm* "belongs to" *Concurrency* (example of type 2 link); and, that *a process* "can have" a *thread* (example of type 3 link). Please, notice here how the type 3 link between *process* and *thread* does not only indicate that the student knows each BC independently. It also indicates that the student has meaningfully learnt a new BC, because s/he is able to correctly relate it to a previous concept.

Finally, it is important to highlight the fleeting nature of a conceptual model. Given that we are continuously learning new concepts that modify our previous assumptions and creating new relationships between already existing concepts and these new ones, the conceptual model is continuously changing and only snapshots of its configuration at a given moment can be modeled.

5.4 Generation of the conceptual and domain model

In this section, the procedure to extract the terms for the domain model and to generate the student's conceptual model for a certain area-of-knowledge is described along with a step-by-step example[1].

To start with the example, let us suppose that we are in the situation of the scenario described in Chapter 1 and, that Sonia is an English Computer Engineering student that this course has enrolled in the Operating Systems subject. She has been assigned Juan as her teacher. Moreover, Juan has chosen that the Operating Systems course will not have any feature as the students are all English and with the same experience in the area-of-knowledge.

5.4.1 The concepts are found

First of all, the teacher has to use an authoring tool to introduce the information of the domain model: the area-of-knowledge to be assessed, the features to be considered for the student model, the topics that this area covers, several questions per topic with their statement, level of difficulty and three or four references. In courses imparted by several teachers, it is desirable that each teacher writes a reference, in order to capture more lexical and syntactic variability.

From the name of the area-of-knowledge, the label of the AC is fixed and, from the name of the topics, the labels of the TCs. In the example, the AC is *Operating Systems* and Juan has said that the topics to cover are: *Introduction, Scheduling, Threads, Processes and Concurrency* and thus, they are the TCs.

Regarding the terms of the domain model that will be the BCs of the conceptual model, they are automatically extracted from the references using a Term Identification module. It has to be trained with a set of balanced samples (50% terms and 50% non-terms). These samples are selected from a domain-specific corpus (i.e. the set of references provided by the teachers) that is compared to a generic corpus (i.e. a set of news on Computer Science retrieved from the web). Each sample has a set of features considered as attributes. They should be at least:

- The relative frequency (freqRel.) of appearance of the term in the domain-specific corpus with respect to its frequency in the generic corpus (i.e. frequency in the specific corpus divided by frequency in the generic corpus). This is because terms tend to be specific to a certain knowledge field and thus, to appear more frequently in the specific corpus and consequently, have a relative frequency higher than one.
- The sequence of part-of-speech (POS) tags of the words composing the sample (e.g. determiner+noun+adjective). This is because terms tend to contain certain POS tags such as nouns, adjectives, etc. but not others such as verbs or adverbs. Moreover, it has been observed that usually terms in the Spanish and English languages can be represented by the following regular expression: NC* NP* ADJ* PREP* NC2* NP2* (zero or more common names, proper names, adjectives, prepositions, more common names and more proper names). Thus, each n-gram extracted from the corpus is matched to the previous

[1]The font type of the example has been changed to facilitate its reading through the main text.

```
word                freqRel  NC   NP    ADJ  PREP  NC2   NP2  Class
if_there_were,      0.2,     0,   0,    0,   0,    0,    0,   0
Dekker algorithm,   1.5,     0,   0.5,  0,   0,    0.5,  0,   1
```

Figure 5.4: Example of training file for the algorithm to automatically extract the terms from the references.

Term	Global frequency in all the references
buddy algorithm	3
Dekker algorithm	1
readers - writers problem	6
Peterson algorithm	1
scheduling algorithm	3
data area	5
atomization	1
control process block	15
memory block	3
process queue	4

Table 5.2: Example of ten terms automatically identified for the example Operating System course and their frequency in all the references provided by the instructors.

regular expression, giving to each of the POS tags a weight equal to the number of words belonging to that class. Later, the weights are normalized so that they all add up one.

Once the training phase is completed, the test starts. The format of the samples is the same but, without the information about whether it is a class or not. It cannot be given in the test file, because this information is the output of the algorithm. Finally, the terms are transformed to their canonical form and, teachers can review this output list. In particular, they can modify it by adding or removing terms as they consider necessary.

The resulting list is stored in the domain model, and as the BCs of the conceptual model of each student. Furthermore, in each student's conceptual model, they are stored with the frequency in which they have appeared in the references and, an initial zero confidence-value to indicate that it is still unknown whether the student knows each BC or not.

For instance, regarding the BCs, if we pick the reference: *"If there were not support from the Operating System, software solutions should be implemented such as the Peterson algorithm and the Dekker algorithm. However, these are quite complex algorithms to implement."*, Figure 5.4 shows the beginning of a sample training file for the algorithm. It can be seen that in the first line of the file, it is written the name of the attributes per each sample and whether it is a term or not (1, yes; 0, no). In the second line, it is written the values for the first trigram found *if_there_were*, which is not a term, and in the third line, it is written the values for the bigram *Dekker_algorithm*, which is a term. Finally, Table 5.2 shows some terms automatically identified for the example on the Operating System course.

5.4.2 The type 1 and 2 relationships between the concepts are found

Type 1 links (AC-TC) are fixed according to the information provided by the instructor. In fact, they are created by connecting the AC with each TC. Thus, they are the same for all the students (although it is important to highlight that the CVs of the concepts are different).

For instance, in Juan's course, five type 1 links have been created between the AC *Operating System* and each one of the five TCs. They form the following propositions:

- "*Operating System* talks about *Introduction to Operating System*".
- "*Operating System* talks about *Processes*".
- "*Operating System* talks about *Threads*".
- "*Operating System* talks about *Concurrency*".
- "*Operating System* talks about *Scheduling*".

Type 2 links (TC-BC) are created between the TC and each BC, whenever a BC is found in a reference of a question of a certain topic. Hence, as above, they are the same for all the students as the references are the same for all of them (although again the CVs of the concepts to be related are different).

For instance, from the sample reference given by Juan "*If there were not support from the Operating System, software solutions should be implemented such as the Peterson algorithm and the Dekker algorithm. However, these are quite complex algorithms to implement.*", "*Dekker algorithm*" and "*Peterson algorithm*" are two sample BCs found. Given that the reference is the correct answer to a question in the *Concurrency* topic, the BCs found in the reference will be related to the TC as:

- "*Concurrency* talks about the *Dekker algorithm*".
- "*Concurrency* talks about the *Peterson algorithm*".

Type 3 links are still not identified in this step as they are automatically extracted from the students' answers.

5.4.3 The free-text ACAA system is used by the student

The first time the students log into the free-text ACAA system, they have to fill in the form for a part of the static component of the student model (e.g. name, mail and preferences). Additionally, as they answer questions, their responses together with the automatic generated scores are also stored in the dynamic component of the student model.

Following with the example, once the domain model is built and, given that Juan has told the class about the possibility of getting more training with an on-line free-text ACAA system, Sonia has decided to try it. The first time that Sonia logs into the free-text ACAA system, she introduces the values of the features chosen by Juan and the session features (next time, she will only be asked about the session features):

- **Name**: Sonia López.
- **Mail**: sonia.lopez@estudiante.uam.es
- **Language**: English.
- **Topics**: Concurrency.
- **Session duration**: Chronometed (1 hour).

- **Feedback**: Detailed.
- **Interface**: Default.

The order in which the questions are chosen is decided by the free-text ACAA system according to the "promotion-demotion difficulty-level procedure". This procedure has been devised to dynamically choose the questions to ask according to how difficult they are, the student's profile and the students' previous answers. It is inspired by:

- The Ausubel's Meaningful Learning Theory, i.e. only when the system detects that easy questions (dealing with a set of concepts) are passed, more difficult questions (dealing with this set of concepts and their related concepts) are asked.
- The good results of CAT systems [Guzmán and Conejo, 2002, Lilley and Barker, 2003] that select the next question depending on how the previous questions have been answered.

The procedure works as follows: during the assessment session, as the student answers the questions of the different topics chosen according to their difficulty levels, the values are modified to adjust the level of the questions to the level of knowledge that each student has in each topic addressed in the area of knowledge under assessment. In particular, the students are promoted to a higher level of difficulty in a topic when they pass the number of questions indicated by the instructor (and stored in the domain model).

On the other hand, when they fail the number of questions indicated by the instructor (and stored in the domain model), they will be demoted to a lower level. Once they are in the highest level, if the students pass the number of questions necessary to be promoted, then the system will consider them as "apt" in this topic. This way, the end-of-session of a free-text ACAA system is not limited just to the number of questions indicated by the student or, to a certain amount of time. It can happen before, whenever the student is considered apt in all the topics chosen.

Finally, it is important to highlight that during the whole process of answering the questions to the free-text ACAA system, students are unaware that their conceptual model is being generated. In fact, they are just focused on answering the questions that the system asks them according to the topics that they have chosen.

For instance, provided that Sonia has chosen to review the "*Concurrency*" topic and that it is the first time she answers a question in the free-text ACAA system, she is asked a simple question (level of difficulty, low, 0) such as: "*What is an operating system?*". If her answer is: *"An operating system is a process with threads."*, then when it is automatically scored by the free-text ACAA system, Sonia may get a 0.1 score over 0.5[2] and thus, as Sonia has failed the question, the next question will be in the same level of difficulty (it cannot be lower than 0). On the other hand, if Sonia gives a correct answer such as: "*It is the interface between the hardware and the user. Some examples are Windows and Linux.*", her response may be given a 0.3 score over 0.5 and she passes the question. Hence, if we suppose that the instructor indicated in the domain model that with one question passed, the student can be promoted, Sonia will be promoted and, the next question for her will be in the next level of difficulty (medium, 1).

[2]The internal procedure to automatically assess the answer and produce the numerical score will be fully explained in the next Chapter.

5.4.4 The confidence-value of each student's concepts is calculated

While the student is answering the questions to the free-text ACAA, the system is keeping track of the use that s/he is doing of the BCs to estimate the confidence-value (CV) in that s/he knows each of them. The core idea is that, the more similar the student uses the BC in comparison to the use of the BC done by the teachers in the references, the better s/he knows the term and thus, the higher the confidence-value is.

To support this idea, the metric should:

1. Include information about the reference knowledge.
2. Include information about the student's performance when answering questions.
3. Take into account that a student's correct answer should be similar to the references.

To give a special relevancy to each of these requisites, a set of metrics have been created to estimate the CV that an individual i knows a certain concept c labeled by a term t, taking as reference knowledge the answers provided by a set of questions Q_i.

In particular, the $ScoreConfidence$ metric is focused on the first and second requisites. It is because it includes the score that the free-text ACAA system gives to the answer (the higher the score, the higher the CV of c labeled by t as the student is correctly using t). In fact, its value is the mean of the weighted scores for the set of questions whose references contain t (i.e. it fulfils the second requisite). Besides, as the weight assigned to each score is calculated as the mean between the frequency of t in the references of the question and all the references, it also fulfils the first requisite.

The $RateConfidence$ metric is focused on the third requisite. That is, it is related to the comparison of the frequency of t in the answer provided by i and the frequency of t in the correct answers taken as references. In fact, it is calculated as the mean of the ratio between the frequency of t in the answers provided by i and the references of all the questions in the area to assess.

Once the metrics have been justified and an intuitive definition has been provided for each of them, it is convenient to formalize these ideas to provide a more precise definition. Please, notice that when lowercase is used, a particular element is referred, while uppercase refers to sets of elements. Let the variables be:

- W, that is the set of words of a language defined as:

$$W = \{ \texttt{ w | w is a word of the language } \} \tag{5.1}$$

- S, that is the set of possible sentences defined as:

$$S = \{ \texttt{ s } | \texttt{ s } \in \bigcup_{n=1}^{\infty} W^n \wedge \texttt{ s is a sentence } \} \tag{5.2}$$

- P, that is the set of noun phrases defined as:

$$P = \{ \texttt{ p } | \texttt{ p } \in \bigcup_{n=1}^{\infty} W^n \wedge \texttt{ p is a noun phrase } \} \tag{5.3}$$

- A, that is an area-of-knowledge:

$$A = \{\ \texttt{s} \mid \texttt{s}\ \in\ \texttt{S}\ \wedge\ \texttt{s is labeled with an AC related to TCs}\ \} \tag{5.4}$$

- T_A, that is the set of terms extracted for A defined as:

$$T_A = \{\ \texttt{t} \mid \texttt{t}\ \in\ \texttt{P}\ \wedge\ \texttt{t labels a BC, a TC or a AC}\ \} \tag{5.5}$$

- Q_A, that is the set of questions to assess A:

$$Q_A = \{\ \texttt{q} \mid \texttt{q}\ \in\ \texttt{S}\ \wedge\ \texttt{q is a question}\ \wedge\ \texttt{q serves to assess A}\ \} \tag{5.6}$$

- I, that is the set of individuals whose knowledge in A is evaluated using Q_A.

$$I = \{\ \texttt{i} \mid \texttt{i is an individual}\ \wedge\ \texttt{i is evaluated using}\ Q_A\} \tag{5.7}$$

- Q_i, that is the set of questions asked to an individual i.

$$Q_i = \{\ \texttt{q} \mid \texttt{i is an individual}\ \wedge\ \texttt{q}\ \in Q_A\} \tag{5.8}$$

Let the functions be:

- answer, provided by $i \in I$ to the question $q \in Q_i$:

$$answer : q_i \longrightarrow \bigcup_{n=1}^{\infty} S^n \tag{5.9}$$

- score, that assigns the numerical mark given by the free-text ACAA system to the answer provided by $i \in I$ to the question q $\in Q_i$:

$$score : answer(q_i) \longrightarrow \Re \tag{5.10}$$

- frequency, for $t \in T_A$ for a set of sentences $s_i \in S$:

$$f : T_A \times \bigcup_{n=1}^{\infty} S^n \longrightarrow N \tag{5.11}$$

- references, for $q \in Q_A$:

$$refs : q \longrightarrow \wp(S) \backslash \emptyset^3 \tag{5.12}$$

From these definitions, Equation 5.13 gives the mathematical formulation of the first metric:

$$ScoreConfidence(t, i, Q_i) = \frac{\sum_{q \in Q_i} score(answer(q)) \times f(t, refs(q))}{f(t, \sum_{q \in Q_i} refs(q))} \tag{5.13}$$

[3]The notation employed refers to the sets of paragraphs in natural language consisting of a set of sentences without considering a paragraph with zero sentences.

Equation 5.14 gives the mathematical formulation of the second metric:

$$RateConfidence(t,i,Q_i) = \frac{\sum_{q \in Q_i \wedge f(t,refs(q)) \neq 0} \frac{f(t,answer(q))}{f(t,refs(q))}}{||Q_i||} \quad (5.14)$$

Given that the range of possible values of $ScoreConfidence$ and $RateConfidence$ are not the same, to make their combination possible, they have to be normalized. It is done by dividing by the maximum value of the range so that both metrics are scaled to the range 0 (minimum confidence in that the concept is known) up to 1 (maximum confidence in that the concept is known).

Therefore, **the metric for CV can be defined as the function that assigns a number from 0 up to 1 to each concept in the conceptual model**. The weight (β) given to $ScoreConfidence$ and $RateConfidence$ depends on which requisite of the metric is considered to be most relevant. It can be fixed (by default) to 50% so that both metrics have the same relevancy. Equation 5.15 is its mathematical formulation:

$$\begin{aligned} CVScore_\beta(t,i,Q_i) = & \frac{ScoreConfidence(t,i,Q_i)}{max_t(ScoreConfidence(t,i,Q_i))} \times \beta \\ & + \frac{RateConfidence(t,i,Q_i)}{max_t(RateConfidence(t,i,Q_i))} \times (1-\beta) \end{aligned}$$

By using this formula, a CV is assigned to each term. Moreover, the BC labeled by this term is assigned a CV. In the case of TCs, the underlying idea is that since BCs belong to one or more TCs, a TC has been understood if most of its BCs have been understood. Hence, the CV of each TC is calculated as the mean value of the CVs of all the BCs related to it. Similarly, once each TC has been assigned a CV, the AC confidence-value is calculated as the mean value of the CVs of the TCs related to it.

As the student keeps answering questions to the free-text ACAA system, the CVs of the BCs, TCs and AC are recalculated. This process of continuously updating the model is crucial in order to be realistic and to model the fleeting nature of someone's conceptual model. Moreover, although a BC has achieved a high CV in an answer, this value could be decreased if this BC is erroneously used in another answer (e.g. the student uses the term that labels this BC many times in his or her answer whereas this term does not appear in any of the references for this question). On the other hand, a BC with a low CV, can get a higher CV if it is correctly used in the following questions.

For instance, let us suppose that the free-text ACAA system has to calculate the CV of the BC *process* for Sonia's conceptual model, and the rest of variables have the following values:

- A = *Operating System*
- $q_1 \in Q_{OperatingSystem}$ = *What is an operating system?*
- $refs(q_1)$ = *An operating system is the application that serves as interface between the user and the processes running by it. A process is a program in execution such as a shell or a database. Examples of operating systems are Windows and Linux.* (Notice that the BC

process appears twice in this reference).

- The BC *process* appears 18 times in all the references of Q_A
- Sonia's answer = *An operating system is a process with threads.*
- Sonia' score for this answer = 0.1 score (over 0.5 as result of the automatic free-text scoring process described in next Chapter).

$$ScoreConfidence(\textsf{process}, Sonia, \{q_1\}) = \frac{0.1 \times 2}{18} = 0.01 \tag{5.15}$$

$$RateConfidence(\textsf{process}, Sonia, \{q_1\}) = \frac{\frac{1}{2}}{1} = 0.5 \tag{5.16}$$

$$CVScore_{\beta=0.5}(\textsf{process}, Sonia, \{q_1\}) = 0.01 \times 0.5 + 0.5 \times 0.5 = 0.255 \tag{5.17}$$

Thus, as can be expected from her partially wrong answer, the CV associated to the BC *process* is quite low (notice that the maximum value is 1). Moreover, as the instructor stated that this BC belongs to the TC *Processes*, its CV is updated taken into account that this TC is linked to two BCs:

$$CVScore(\textsf{Processes}) = 0.255/2 = 0.18 \tag{5.18}$$

The same is applicable for the CV of the AC, taken into account that this AC is linked to five TCs:

$$CVScore(\textsf{Operating Systems}) = 0.18/5 = 0.04 \tag{5.19}$$

As it is expected, the area-of-knowledge *Operating Systems* is quite big and, a low CV is expected when it is calculated just from one concept of one question.

5.4.5 The type 3 relationships between concepts are found

Type 3 links (BC-BC) are automatically extracted from the answers provided by the students according to the following procedure:

1. Find the first BC and mark it as the first BC of the relationship.
2. Find another BC in the same sentence and mark it as the second BC of the relationship.
3. Extract the words between the first and the second BC and, mark them as the linking words of the relationship.

In the above example answer given by Sonia, *"An operating system is a process with threads"*.:

- First BC = *process*

- Linking word = with
- Second BC = *threads*

5.4.6 The conceptual model is updated by instructors, the free-text ACAA system and/or students

It is important to highlight that this procedure is cyclic, starting with the information provided by the teacher and by the answers of the students, to generate the initial conceptual model that is reused again to update itself according to the new use of the concepts in the new answers provided by the student. Moreover, that the conceptual model is not only useful to be displayed to teachers and students, but to the own system to decide which questions should be asked to the students as the terms that they cover have a low CV in the conceptual model, that again is modified with the new answers provided.

Instructors and students should have free access to see the conceptual model generated (i.e. the instructor can see each student's conceptual model and each student his or her own conceptual model). That way, instructors can look at how the concepts have been understood up to each point of the course and how to remedy the lack of knowledge and misconceptions discovered, and students can organize their study by identifying which their strong and weak points are.

The following main knowledge errors about an area-of-knowledge can be detected from each student's conceptual model:

- **For concepts**:
  - *Ignorance*: All concepts (irrespectively of their condition as AC, TC or BC) that have been associated a CV of zero, are unknown by students as they have never used them (even in questions in which they are needed as these terms have been highly used by the instructors in the references).
  - *Misconceptions*: Some concepts may seem to be known by students as sometimes they use them. However, they are not completely understood as in some occasions they have used them wrongly in their answers.
- **For links**:
  - *Ignorance*: Type 1 (AC-TC) and 2 (TC-BC) links are fixed according to the hierarchy given by the instructors. However, it can be indicated that a student does not know a type 1 or 2 link whenever the concepts that they have at their extremes are unknown. Regarding type 3 (BC-BC) links, if they are not in the student's answers, then they do not appear in the conceptual model and it may indicate that the student ignores them.
  - *Erroneous links*: Sometimes the type 3 link appears connecting two concepts that should not be connected. This evidences an error in the cognitive structure of the student as s/he believes that they are related in a wrong way. It is fundamental to correct this situation to allow him or her to continue learning meaningfully and linking correctly new BCs to the previous BCs (as Ausubel claimed in his theory of

Meaningful Learning, see Section 2.5).

It can be seen that type 3 links are very important as they reflect how the concepts are related in the cognitive structure of the student. Type 2 are also important since they give us information about how the basic-concepts are grouped into topic-concepts and, how well each topic-concept is understood. From a BC, several type 2 links to TCs with a certain strength can be created. The higher the strength, the more relevancy that the BC has for this topic.

The free-text ACAA system also uses the conceptual model to foster reflective thinking among students so that they do not just answer in blank to get the references. It is because the free-text ACAA system main goal is formative, and thus, students who fail a question should not have direct access to the references provided by the instructors. It would be better to guide them towards the correct answer. Moreover, to guide students according to how they use the concepts in their answers. That way, it is possible to find out the reason why they do not use some BCs in their answers. For instance, if it is because students ignore the BC, they will not be able to answer a specific question about it. On the other hand, if students know the BC but they have forgotten to use it, they will be able to correctly answer the specific question. Hence, a procedure of asking extra questions that can be called "clarification questions" or "compensation questions" has been devised. The questions proposed in this work are:

- **The first level clarification question**: It is in the form *"Please, explain your previous answer more"*. It is simple but powerful since many students know the answer and they do not need to be given extra hints to solve the question. The problem is that they do not make the effort to express all the information they know and they just want to pass to the next question even leaving fields in blank [Aleven et al., 2004].
- **The second level clarification question**: It is in the form *"Please, explain more about X"* where X is the BC related to the original question with a confidence-value lower than a certain threshold indicating that the student does not use this concept.
- **The third level clarification question**: It is in the form *"Is it true that Y?"* where Y can be a sentence extracted from the references and thus, it is expected an affirmative answer from the student to pass the question. Moreover, to avoid that students learn that they always have to answer *Yes*, since they have found out that Y is extracted from the references, the system can also automatically negate Y. That way, it is expected a negative answer from the student to pass the question. The procedure of automatic negation is completely random, and focused on the verbs and adjectives of Y:
  - For verbs, it can add a "*no*" particle before. For example, "*Is it true that Unix is an operating system?*" will be transformed into "*Is it true that Unix is not an operating system?*".
  - For adjectives, it can use the antonym as provided by WordNet [Vossen, 1998]. For example, "*Is it true that FCFS is the best short term scheduling algorithm?*" will be transformed into "*Is it true that FCFS is the worst short term scheduling algorithm?*".

The free-text ACAA system should reevaluate the new answer given by the student to each compensation question (taking also into account the information given in previous answers) so

that if the score continues being less than the minimum required to pass it, the next compensation level question is presented. The process stops when the student is able to pass the question or the third level compensation question is failed.

As the CVs associated to each concept in the student's conceptual model are recalculated each time the student answers new questions, the clarification questions are also updated according to the new CVs. For instance, if a student uses correctly a BC in the new questions that s/he passes, then the next time s/he fails a question, this BC will not be presented in the second level clarification question. This way, the focus is on the BCs that are still unknown.

An example dialog could be as follows (as in the example shown at the beginning of Chapter 2, concepts are marked in bold):

Question What is an **operating system**?

Sonia's answer It is a **process** with **threads**.

Question Please, explain your previous answer more.

Sonia's answer It is the first **application** executed in the **computer**.

Question Please, explain more about **application**.

Sonia's answer The **operating system** is an **application** that serves as an interface between the **hardware** and the **user**.

Question Is it true that **Unix** and **Windows** are examples of **operating systems**?

Sonia's answer Yes.

In this example, as the student does not know the TC *Introduction to Operating Systems*, a question involving BCs belonging to this TC appears. Let us suppose that the maximum score for this question is 0.5 and it is necessary a 0.3 to pass it. The first answer given by the student ("*It is a process with threads.*") receives a 0.1 score so she fails and the clarification dialog starts. The first level clarification question is presented and the new answer is concatenated to the previous answer so now the answer is "*It is a process with threads. It is the first application executed in the computer.*" and it receives a 0.2 score. It is better but still not enough as it is unclear that Sonia understands the concept *application*.

Thus, the second level clarification question is presented, and Sonia is directly asked about this concept. The new answer is correct and Sonia gets a higher CV for the BC *application*. However, she is unable to pass the question because her answer lacks examples of operating systems. Finally, the third level clarification question is presented directly asking for the information needed. Now the text concatenated to the answer is not "*Yes*" but the sentence asked to the student. That is, the complete answer given by the student has been "*It is a process with threads. It is the first application executed in the computer. The operating system is an application that serves as an interface between*

the hardware and the user. Unix and Windows are examples of operating systems.". It receives a 0.3 score and the student passes the question.

It is important to highlight that not only the dialog is generated according to the conceptual model, but the conceptual model is modified according to the dialog. For instance, in the previous example, each time a new BC has been mentioned, its CV has been recalculated. Moreover, new type 3 links have been extracted. In particular, the CVs of the BCs *operating system*, *application*, *computer*, *hardware*, *user*, *Unix* and *Windows* have been updated and two new type 3 links have been found "*application executed in the computer*" and "*operating system serves as an interface*".

5.5 Class conceptual model

Although the focus of student's modeling has been on individuals during many years, recently, emerging applications have also started to consider generic models of user communities [Webb et al., 2001, Ungar and Foster, 1998]. This is very important because by using a class conceptual model, not only the particular misconceptions of a student are identified, but the misconceptions of the whole class. In fact, in the class conceptual model, the confidence value associated to each concept indicates how the whole class has understood the concepts exposed in the lesson.

The goal here is not to provide adaptation to each particular student, but to help the teacher to organize the syllabus of the course. That is, to identify which concepts are missing in the class conceptual model and thus, which concepts should be introduced in the lesson before attempting to study the next lessons. Moreover, to analyze which concepts from the ones explained to the class have been assimilated and if they have correctly been linked to the previous ones. Students can also access the class conceptual model to compare it with their own conceptual model and, to find out which links have been erroneously introduced and which ones are missing.

Equation 5.20 shows the mathematical formulation of the first metric for a set I of students:

$$ScoreConfidenceGroup(t, I, Q_I) = \sum_{i \in I} \frac{ScoreConfidence(t, i, Q_i)}{||I||} \tag{5.20}$$

Similarly, Equation 5.21 is the mathematical formulation of the second metric, and Equation 5.22 is the mathematical formulation of the CV to be assigned to each BC of the whole class conceptual model:

$$RateConfidenceGroup(t, I, Q_I) = \sum_{i \in I} \frac{RateConfidence(t, i, Q_i)}{||I||} \tag{5.21}$$

$$CVScoreGroup(t, I, Q_I) = \sum_{i \in I} \frac{CVScore(t, i, Q_i)}{||I||} \tag{5.22}$$

TCs and ACs confidence values for the group conceptual model are calculated as in the particular student conceptual model but from these the values achieve with Equation 5.22 so that the results are common to all the students. Please, notice also that for the creation of type 3 links, not only the answers of one student, but all the students' answers are processed.

Chapter 6

An example of free-text ACAA system: Willow

Willow is a free-text Adaptive Computer Assisted Assessment (ACAA) system. That is, it is able to automatically score answers written by students in free-text and to tailor the assessment to each student's model. Willow is an enhanced version of the free-text CAA Atenea system explained in Section 6.1. Willow keeps all Atenea's features and adds new ones that are explained in Section 6.2. The main goal of Willow is not to replace teachers or to address all type of questions, but to provide an alternative form of evaluation that focuses on formative assessing open-ended questions. Figures 6.1 and 6.2 show two sample snapshots of Willow.

Moreover, Willow is the key system in the framework of the Will tools to automatically generate the students' conceptual models from the answers provided to the system. It implements the procedure described in the previous Chapter, by keeping track of the terms used in the students' answers, estimating the CV of each concept and extracting the links between the concepts. Additionally, Willow has associated an authoring tool called Willed that helps teachers to introduce the information about the area-of-knowledge to build the domain model as explained in Section 6.3.

The core idea of Willow is the common assumption that "*the more similar the student's answer is to a set of correct answers, the better it is, and thus, the higher the score the answer achieves*". In order to measure the similarity between the student's answer and the correct answers or references, Willow uses a combination of statistical techniques such as n-grams and Latent Semantic Analysis (LSA) and other NLP techniques that are included in the wraetlic tools [Alfonseca et al., 2003, 2006] [http22]. Sections 6.4 and 6.5 explains the high and low level architecture of Willow. Finally, Section 6.6 details which is the optimum use of Willow.

6.1 Non-adaptive version of Willow: Atenea

Atenea is an on-line system able to assess free-text students' answers automatically. Atenea has served as bases for Willow. In fact, both, Willow and Atenea, use the same assessment engine but Atenea does not use any AH technique. That is, Atenea randomly chooses the

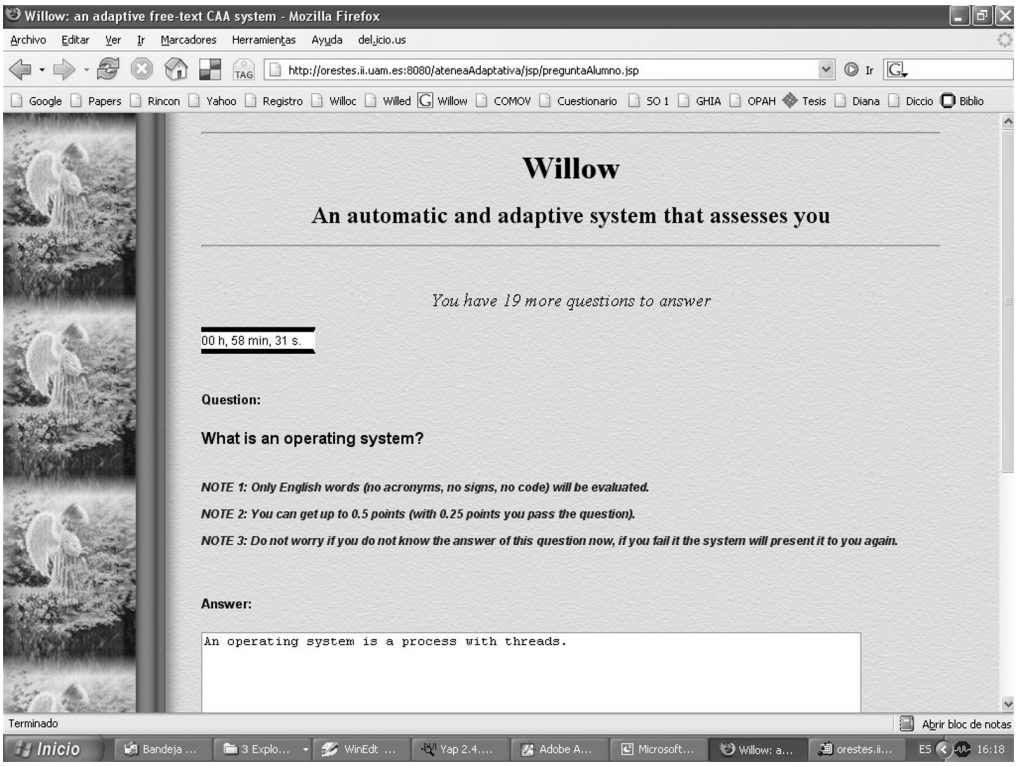


Figure 6.1: A question delivered by the system according to the settings provided.

Atenea	**Willow**
The same statements for all the students	Different versions of the statements
The same references for all the students	Different versions of the references
The next question is randomly chosen	It is chosen using the promotion-demotion procedure
It does not keep a student's model	It stores both static and dynamic information about the student
The same interface for all the students	A personalized interface
Fixed end-of-session	Dynamic end-of-session
It does not foster reflective thinking	It guides the students toward the correct answer
It does not generate conceptual models	It generates individual and group students' conceptual models

Table 6.1: Main differences between Atenea and Willow.

next question to ask irrespectively of the student's preferences. It has the same interface for all students and does not keep track of the necessary information to generate the student's conceptual model. Table 6.1 gathers the main differences between both systems and Figure 6.3 shows some snapshots of the system.

Atenea can be used as a stand-alone system or integrated in AEHSs such as TANGOW [Carro et al., 1999] according to the following protocol:

- The AEH system gathers the information about the student's profile and sends it to Atenea: the student login, age, language, experience, the dataset that is being studied, the feedback volume and the number of exercises to be assessed.
- Atenea randomly chooses a question of the dataset, that has not already been solved by the student (that is, not yet graded or graded less than half of the maximum score).

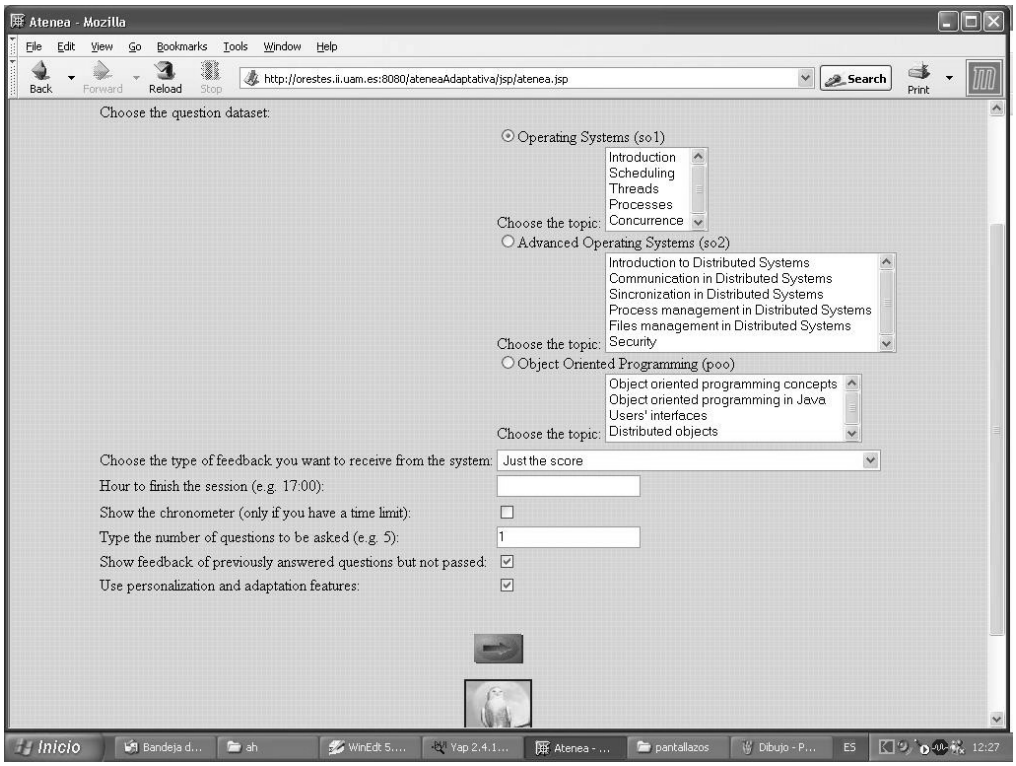


Figure 6.2: A snapshot of Willow's configuration session stage.

System	Domain	Availability
Willow	Computer Science	Academic

Table 6.2: Domain in which Willow has been applied and its availability.

The question is chosen taking into consideration the student age and experience. After the students' answers submission, Atenea evaluates them and returns the score and the feedback to the student. This process is repeated until the student has answered the number of exercises demanded. In the end, when this stop condition is satisfied, Atenea returns to the AEH system the student login and information about which questions have been asked and the score the student has achieved in each one.

6.2 Willow's main features

Tables 6.2 and 6.3 gather the values of the features for the case of Willow in the tables of comparison of the state-of-the-art in Chapter 4. Moreover, Willow has been provided with the following features in order to live up to nearly all the lecturers' expectations found by Darus and Stapa [2001] (shown in Table 4.2) ordered from more to less importance, Willow:

- **Indicates errors and marks organization of ideas**: Willow keeps track of the students' answers to generate the students' conceptual models and show them to the instructors and the students. In this way, teachers can identify some misconceptions and

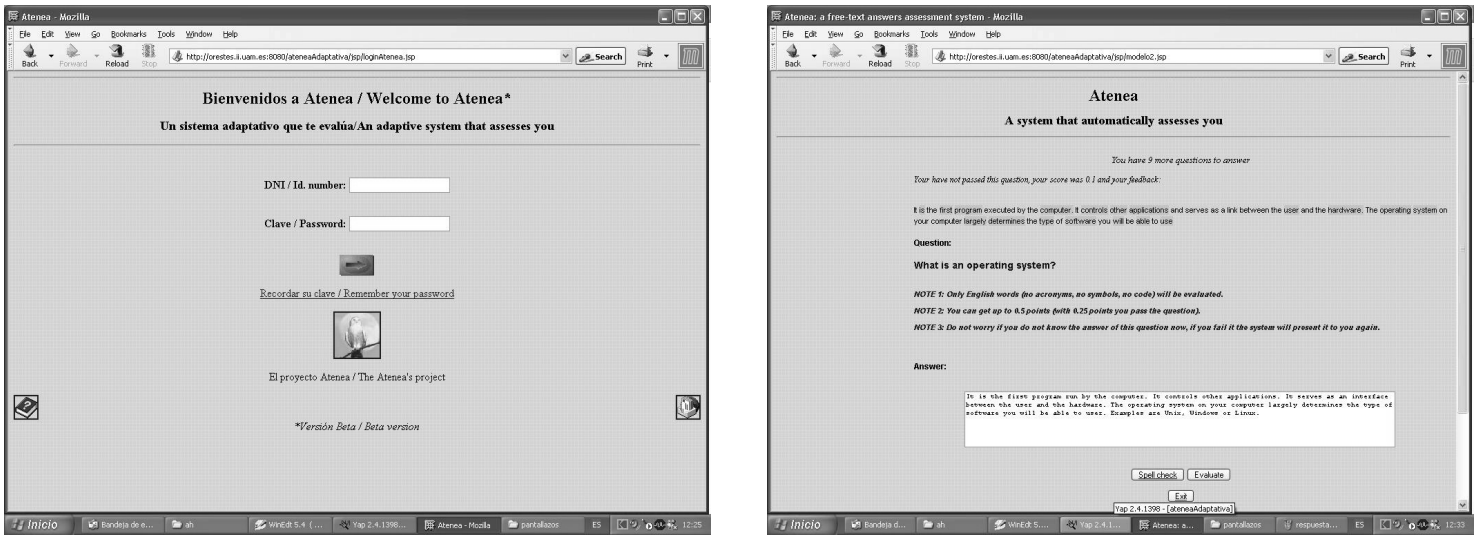


Figure 6.3: A snapshot of the login page of Atenea and of page delivering a question.

System	Technique	Evaluation	Language
Willow	NLP+AH and statistics	Corr:.54	Spanish
Willow	NLP+AH, LSA and statistics	Corr:.56	English

Table 6.3: Techniques used in Willow, languages that it can process and its result measured as the Pearson correlation between the teachers' and Willow's scores for the corpus used as explained in Chapter 8.

erroneous links among concepts that students have. More information about the feedback given to the students can be seen below.

- **Marks syntax**: The possibility of doing partial parsing is included. However, the focus is more on the concepts and their relationships.
- **Provides error statistics**: A rank of the most important concepts that students should know (as automatically identified by the Term Identification module and modified or not by the instructors) is kept. In this way, Willow can give information about how the most important terms are known, and calculate other statistics such as the concepts that are best understood.
- **Marks non-native speakers writing**: At present, Willow is able to process answers written both in Spanish and in English. Nevertheless, it is important to notice that Willow is not aimed to be a Computer Assisted Language Learning system [Catt and Hirst, 1990].
- **Produces a grade**: The score produced by the internal modules of Willow is always between 0 (very bad answer) and 1 (very good answer). However, when it is shown to the lecturer, it is scaled to the range indicated. For instance, if the lecturer has given as maximum score for a certain question a value of 0.5, then the score provided by Willow is scaled to the range [0, 0.5] according to the procedures explained later in Section 6.5.3. It could also be easily shown as a letter grade, just by asking lecturers to give the categorization to apply (e.g. scores between 0-5, F; between 5-6, E, and so on).

- **Marks surface features**: Willow takes into account some features of the student's answer such as the vocabulary employed, its variability and the length of the answer. This is because one of the techniques that it uses is Evaluating Responses with Bleu (ERB) that depends on the length of the text to process and focuses on the words and combination of words employed.
- **Gives individual feedback**: Willow generates a feedback page for each student not only with the score achieved in the question, but also with the processed text as corrected by the NLP techniques used in Willow and the correct answers provided by the instructors. Moreover, this page is only shown if the student has passed the question or has completed the set of clarification questions to guide him or her to the correct answer.
- **Marks holistically**: Unlike other existing free-text CAA systems that give a different score per style, content, etc. of the answer, Willow gives a unique score (holistic score) that reflects all aspects of the answer. Nevertheless, it could also be possible to give a separate score for the content of the answer (as given by LSA) and the style (as given by ERB).
- **Marks knowledge content**: According to Landauer et al. [1998], LSA is able to capture the internal relationships between words according to their semantic relatedness. In this way, an attempt to take into account the semantics and coherence of the answer is also addressed.
- **Marks analytically**: Automatic assessment of free-text answers is by nature systematic. It is because it is done with a computer and thus, provided that the input text is the same, the output score should also be the same. In fact, it has been tested that Willow fully fulfils this point.
- **Marks according to disciplines**: The hierarchical structure of knowledge in Willow permits to work with different values of the evaluation variables such as maximum scores per question, percentage to consider that the question is passed, etc. depending on the area-of-knowledge under assessment.

Regarding the students' expectations, it is important to remember that, according to Darus et al. [2001], they were all focused on the feedback. Thus, what is considered here is the areas of feedback expected by the students (see Table 4.3) ordered from more to less relevancy. In particular, Willow is able to give feedback to the students about their:

- **Errors, domain knowledge and organization of ideas**: The student's processed answer is shown to the students with the best points (as they match the references) marked in green and the rest of the text, marked in grey. In this way, students can easily identify the irrelevant information provided in the answer (that may include errors and in any case, should not be present). See Figure 6.4 for an example of Willow's feedback page. Besides, students can inspect the conceptual models as generated by Willow to see which concepts have been assigned a low CV as they have not been used when required or have been used wrongly in contexts where they should not have been used. Students can also detect erroneous links in their conceptual models as they can compare them with the class conceptual model.

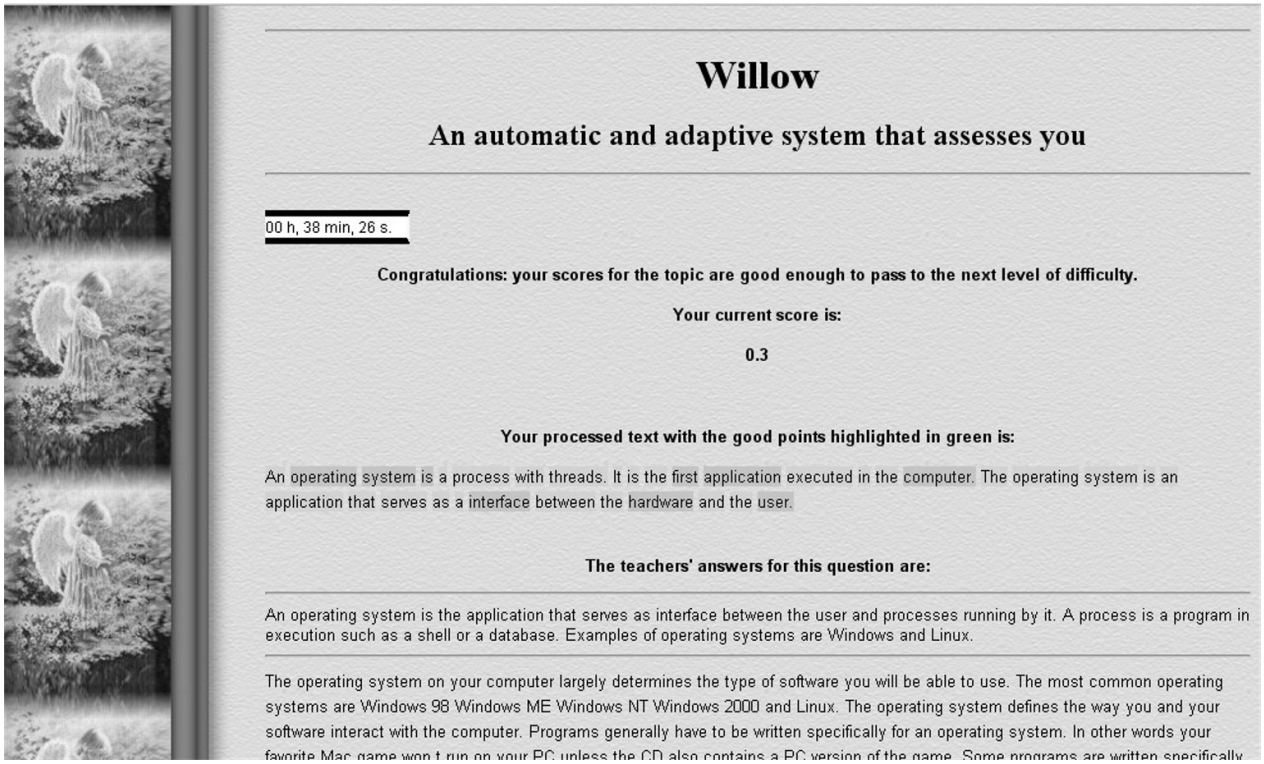


Figure 6.4: A snapshot of Willow's feedback page.

- **Coherence of text**: The score given to students is calculated as the combination of the ERB and LSA score. Given that, according to Landauer et al. [1998], LSA can be used as an indicator of the coherence of the text, the holistic score provided includes also a measure of the coherence of the text. Moreover, students can see their LSA score for each answer.
- **Their English proficiency**: This point can be ignored in our case as students are not forced to write in English in order to use the system because it is not a Computer Assisted Language Learning tool.
- **Creativity**: This can also be ignored as the domain to which Willow should be applied must remain objective and comparable to a set of references provided by the instructors.
- **Style of writing and syntax**: The style of writing can be captured by using the statistics metrics included in Willow such as ERB and the tools included in the wraetlic package.

The only feature that is not included in Willow is the one regarding the identification of the rhetorical structure of the answer and to give students and teachers feedback about it. A preliminary experiment was done by manually identifying different points of the students' answers such as when they were giving advantages, disadvantages, etc. and it was found out that the task is utterly complex (at least for Spanish answers). Students tend to mix all the points in very long sentences (that in many cases are not grammatically correct). Thus, this option remains discarded until more NLP techniques such as the ones proposed by Marcu [2000] can be adapted to be used by the system.

On the other hand, Willow offers other extra possibilities such as:

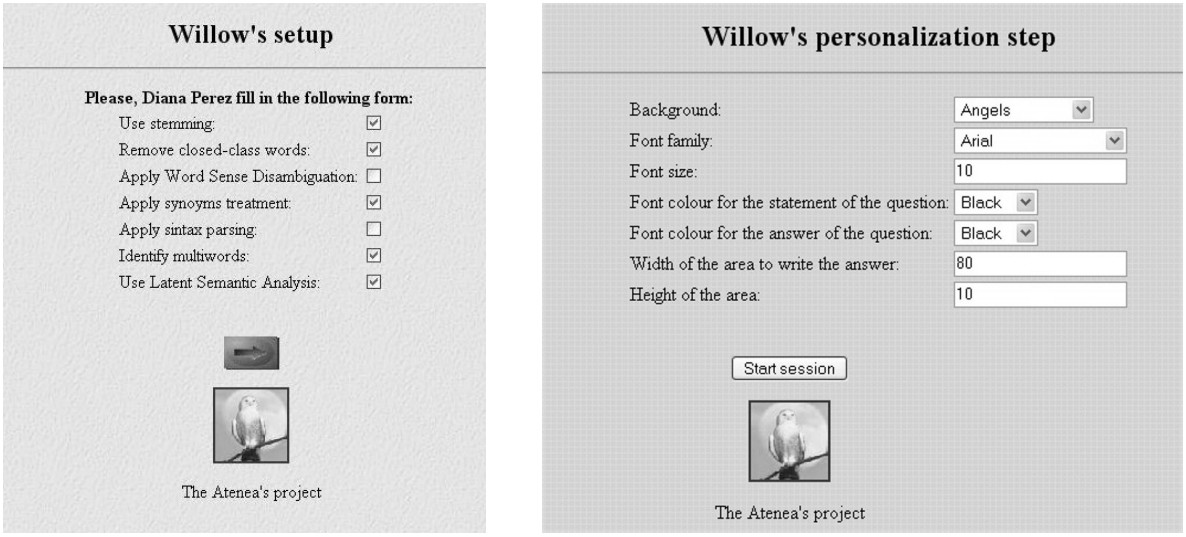


Figure 6.5: On the left, a snapshot of Willoc and on the right, the Willow's personalization possibilities.

- **On-line registration**: Any individual that wants to use Willow only needs to fill in an on-line form providing the following data: name, identification number, password, year and mail. This information will be stored in the student model. The registration only needs to be done once.
- **On-line configuration**: As Willow relies on several NLP techniques to process the student's answer and references and make them more comparable, the administrator of the system is given the possibility of choosing which techniques are to be applied. S/he can do it on-line by using the configuration tool called Willoc (see Figure 6.5 left). If none is chosen, then only the statistics module based on ERB will be applied (necessary for the final comparison between the students' answer and the references). It is also possible to only choose LSA and thus, both ERB and LSA will be used.
- **On-line help**: There is the possibility of reading about Willow and how to use it from a link in the own system and from the contextual information of certain items.
- **Indicators of progress**: The end-of-the-session can be chosen as a number of questions to complete or a certain amount of time. Hence, there are two main indicators of progress: the number of questions remaining (e.g. as shown in Figure 6.1: "*You have still to answer 9 more questions*") and the remaining time. In any case, the student can log out the system at any time pressing the "*Exit*" button and the system can consider that the session is over, provided that the student has finished all the questions according to his or her level of knowledge.
- **Previous feedback reminder**: In the case that a student (even having completed the set of clarification questions) is unable to pass a question, Willow marks this question as failed and to be asked again in the future. Moreover, students are given the possibility of being shown the feedback that Willow provided them the last time that they failed the question. In this way, they are warned to fix the mistakes and are given a new chance to

pass the question. See in Figure 6.3 how this feature (that was already implemented in Atenea) is shown to the student.

- **Emotional interaction**: Due to the problem, reported in the literature with on-line educational systems, that the student could feel isolated, Willow has several mechanisms to communicate with the student not only with the standard feedback (i.e. the score, the processed text and the references) but also with sentences such as "*Congratulations, you can pass to the next level*" when they have passed all the questions of a level of difficulty or "*Your scores are not good enough to be in this level, you go down a level*" that attempt to emulate a game in which the student has to complete a set of levels. Besides, if s/he does not agree with the feedback provided, then s/he has the option to make a complain and asks the teacher for a clarification of the statement of the question when s/he is in a medium or lower level of difficulty (more advanced students should understand the questions without being helped).
- **Full control over the interface**: The student can modify all the elements of the interface as s/he wishes: the background, the font type, the size of the font, the statement and answer color, etc. See Figure 6.5 (right).
- **Spell checking**: Sometimes when typing, mistakes are committed not because the correct spelling is unknown but because of the use of a keyboard as the input method. These mistakes should not be considered in the evaluation and thus, students are given the possibility of using a spell checker to correct them.

Finally, it is important to mention that Willow has two restrictions regarding the areas of knowledge to which it can be applied: it should not be applied to assess creativity or subjective opinions as there is no correct answer to which to compare these issues; and, neither mathematical content nor code is processed as Willow's focus is on processing free text.

6.3 Willed

Willow has an associated authoring tool called Willed. In Willed, the teacher has to select the area-of-knowledge to use (or to create a new one) and, which question to modify or to add a new one. For each question, the teachers have to write a statement and a set of references. Besides, they have to associate the question to a topic and a level of difficulty inside the topic (e.g. low, 0; medium, 1; or, difficult, 2). Willed has also on-line help to orient its users.

6.3.1 Creation of a new area-of-knowledge

In Willow, a course about a certain area-of-knowledge is always associated to a collection of questions comprising several topics (at least one question belonging to one topic). Moreover, in order to allow the adaptation to each users' preferences, teachers are asked to introduce a set of features that should be considered to tailor the assessment to the student. There are three optional features predefined, as they are the most common: language, age and experience. Teachers can select them, add new ones or not select any. See Appendix A for more technical

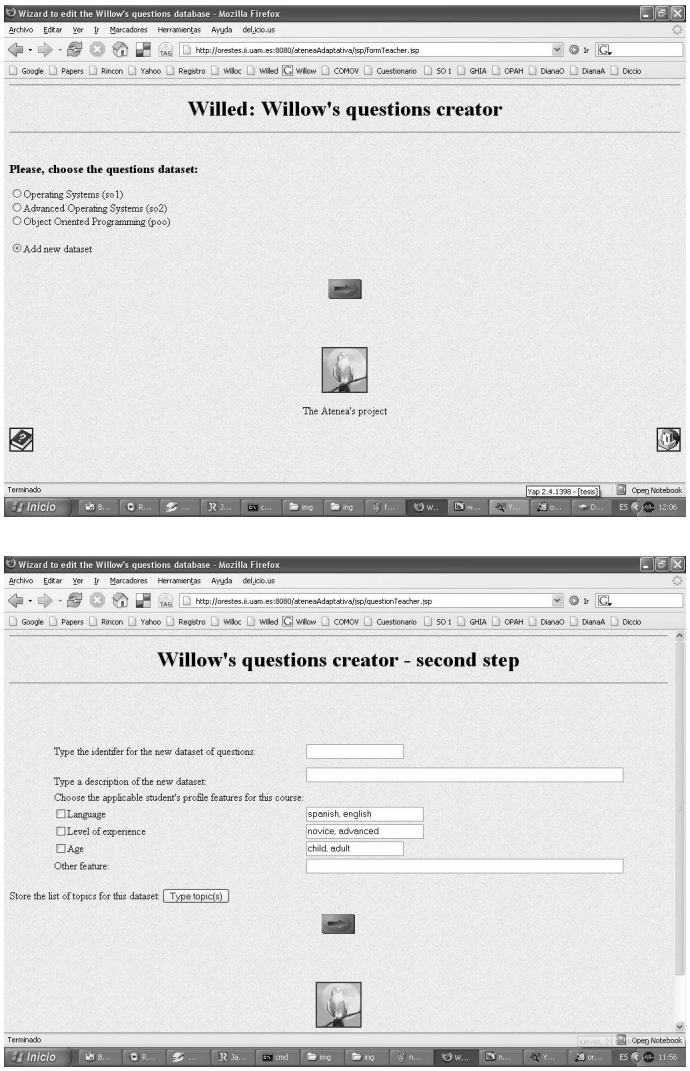


Figure 6.6: On the top, a snapshot of the Willed's page with the initial menu (to modify an already area-of-knowledge collection of questions or to create a new one) and at the bottom, the page to create a new area-of-knowledge.

details. Figure 6.6 (top) shows a snapshot of Willed's menu and, Figure 6.6 (bottom) shows a snapshot of Willed's page to create a new area-of-knowledge.

Willed uses an algorithm that automatically combines all the values of the features indicated per area-of-knowledge and asks the instructor to fill in the information about each one. That is, if the features chosen are language (Spanish or English) and experience (novice or advanced), then Willed will ask the teacher for the statement of the question for a novice Spanish, advanced Spanish, novice English and advanced English. The same is applicable for the references. Moreover, in order to make this process easier for instructors, the translation is supported with the Machine Translation engine of Babelfish [http23] to have a preliminary translation of his or her answer to other language (teachers can modify this translation to improve it).

Furthermore, a Term Identification module based on the technique described by Ballester et al. [2002] has been implemented so that no teacher has to introduce by hand the list of

most relevant terms of the area-of-knowledge. On the other hand, the terms are automatically identified from the references using the C4.5 algorithm. This algorithm is able to learn a decision tree [Quinlan, 1986] to decide whether each n-gram in the references is a term or not. As with the automatic translations, teachers have the possibility of modifying this list of terms. Finally, in order to complete the creation of a question, teachers have to indicate the maximum score that a student can get in the question if her or his answer is perfect, the threshold score so that Willow considers that the student has passed the question, the topic to which the question belongs and its difficulty level.

6.3.2 Modification of an existing area-of-knowledge

If the course is not being used, all the values fixed at the creation stage can be modified: the name, the description and the features (addition, removal and modification). However, as indicated and to avoid inconsistencies, once a course is running, teachers are not allowed to modify or remove them. On the other hand, they are allowed to introduce new topics and questions. See Figure 6.7 (top).

Nevertheless, even when the course is running, teachers can add new questions as shown in Figure 6.7 (bottom). Moreover, they can change the statement, references, topic and difficulty level of any question. This is because sometimes these modifications are necessary in order to allow a better assessment. For instance, a statement could be wrongly formulated, a reference could have a mistake, and it could be possible that a question that may have seemed easy for an instructor is not so easy for the students. Thus, if the course starts and the students fail a question massively, its difficulty level should be changed to a higher value. Figure 6.8 (top) shows a snapshot of Willed's page to modify an existing question. Teachers can also add more references to each question as shown in Figure 6.8 (bottom).

Furthermore, given that references are crucial to the assessment because Willow's core idea is to compare the student's answer to them, teachers can be helped so that all the responsibility of creating the references is not only theirs, but an automatic procedure is able to automatically generate new references from the ones provided by the instructor.

This is done in Willed using Anaphora Resolution (AR). AR is the process of finding the antecedent of an anaphora. For example, in the fragment *Unix is an operating system. It is multiuser*, the anaphora is between the anaphor *It* and the antecedent *Unix* as they refer to the same entity. It is said that *It* and *Unix* belong to the same coreferential chain.

The core idea to automatically generate new references using AR is to replace the noun phrases (NPs) in the coreferential chains with other referential entities of those NPs. For instance, if we consider that the fragment *Unix is an operating system. It is multiuser* is a reference written by a teacher, two new references can be generated from the coreferential chain [Unix,an operating system,it]:"*Unix is an operating system. Unix is multiuser*" and "*Unix is an operating system. An operating system is multiuser*".

In order to achieve this result, the AR-Engine RARE [Cristea and Dima, 2001] has been used to find the coreferential chains in the references. In particular, three techniques have been

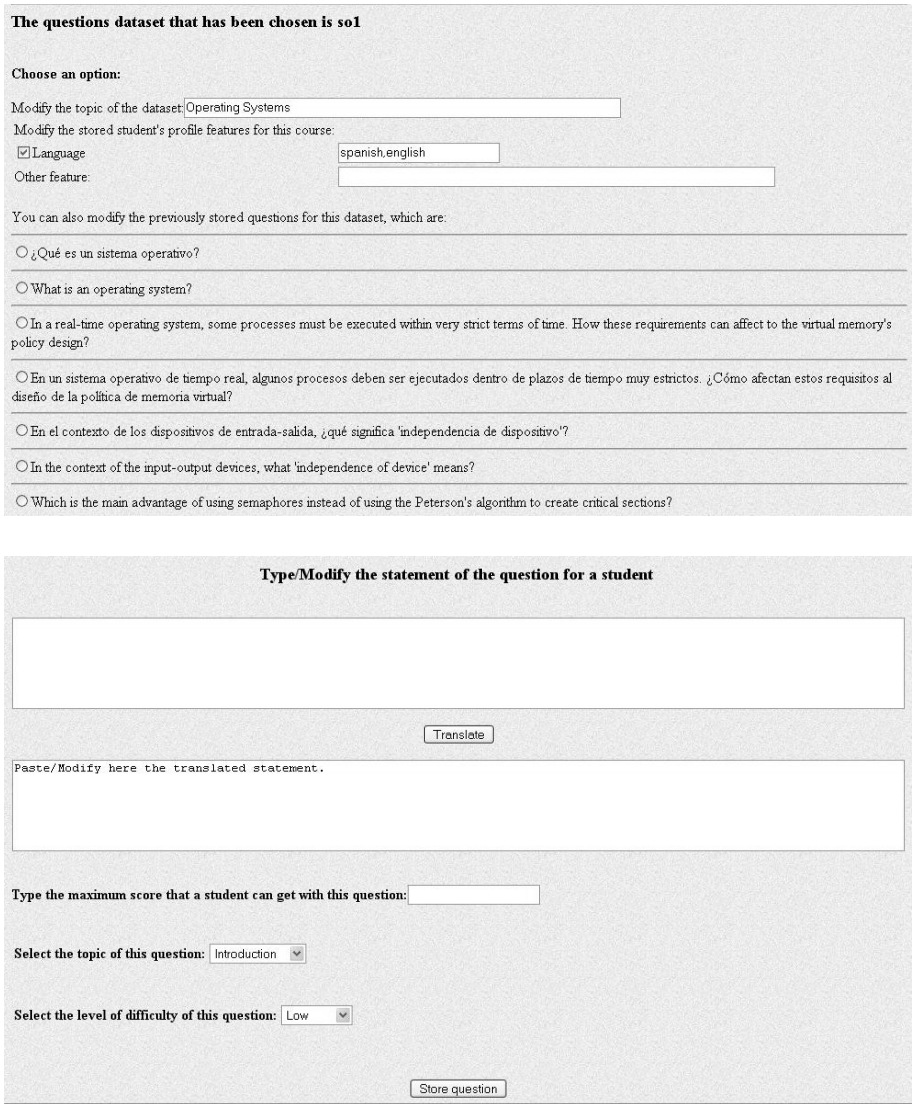


Figure 6.7: At the top, a snapshot of the Willed's page to modify an already existing collection of questions and at the bottom, the page to create a new one question.

devised: *First-NP*, in which each NP in the coreferential chain is substituted for the first NP which is not an "*it*"; *All-NPs*, in which each NP in the references is substituted for the whole coreferential chain to which it belongs; and, *Only-it*, in which only non-pleonastic[1] *it* pronouns in the references are substituted for the first NP in the coreferential chain which is not an *it*.

Finally, to implement the procedure for automatically generating new paraphrases of the references, the following pseudocode has been used. It starts with one reference text that has been written by hand by a teacher.

1. `Initialize an empty array` $genRefTexts$ `with the reference text.`
2. `Look for the next non-pleonastic ‘‘it’’. If none is found, stop.`

[1] A pleonastic pronoun is a pronoun that cannot be replaced by a noun phrase as it has not lexical meaning. For instance, in the sentence: *It is obvious that ...*, the "*It*" is pleonastic as it does not refers to any particular subject.

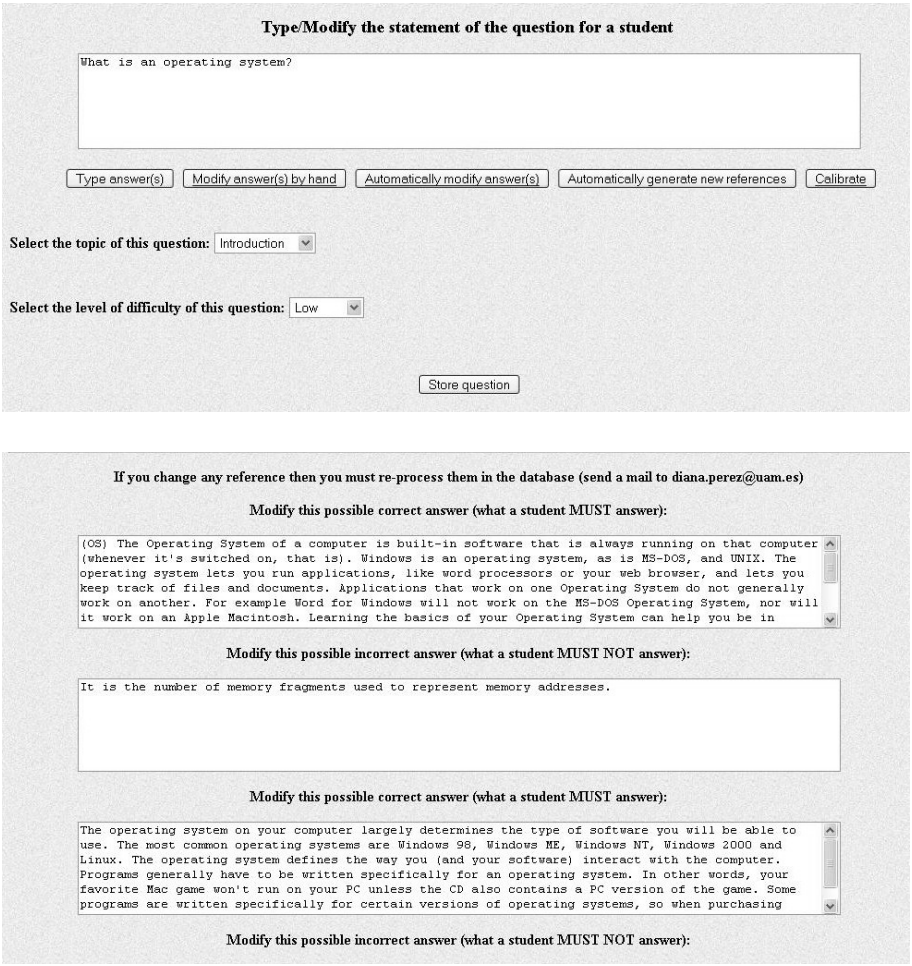


Figure 6.8: At the top, a snapshot of the Willed's page to modify an already existing question and at the bottom, the page to modify its references.

3. Identify the row of the table that contains the coreferential chain which includes the ‘‘*it*’’ pronoun found.
4. Create as many copies of all the references in $genRefTexts$ as NPs exist in the chain. For each of the copies, the last ‘‘*it*’’ found has been replaced by each possible RE.
5. Go back to the second step.

Figure 6.9 shows an execution example. This process is performed off-line by the system as it is quite time consuming. Thus, whenever teachers request in Willed to have new automatically generated references, Willed sends a mail to the administrator of the system to run it and introduce the new references for the course.

6.4 High-level architecture

Figure 6.10 shows an overview of the high-level architecture of Willow. As can be seen, the input to Willow is a student's answer in plain text, the knowledge contained in the domain

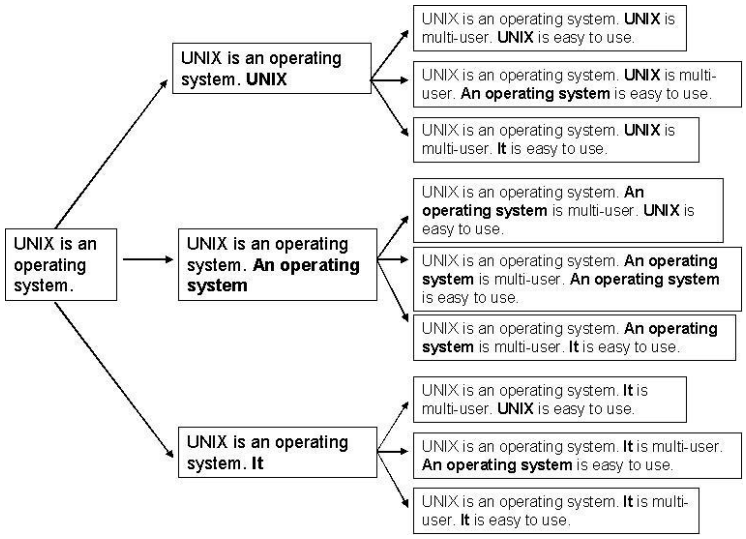


Figure 6.9: Example of the generation of new references from the original text "*Unix is an operating system. It is multi-user. It is easy to use*".

model as stored with Willed, the student models as stored in the database and, as external lexical resources: WordNet 1.7 for English and the Spanish EuroWordNet for Spanish [Vossen, 1998]. WordNet is a semantic network of words in which words can be looked up conceptually instead of alphabetically. Both WordNet and the Spanish EuroWordNet have been used as processed by Alfonseca [2003].

Given that Willow is based on the combined used of Adaptive Hypermedia and Natural Language Processing techniques to automatically and adaptively assess the students' answers, in the Willow's high-level architecture some modules implement the AH techniques and others, the NLP techniques. In particular, the main module for the AH techniques is the **Question planner** that chooses which question should be asked next and how (i.e. suitable version of the statement and references) according to the stored student model. Moreover, it also manages the set of clarification questions. Willow's interface is also modified according to the values defined by the student (if any, otherwise, the default interface is shown). On the other hand, the modules for the NLP techniques are the following:

- **The processing module** that aims to make the student's answer and the references more comparable transforming them into a more manageable format. In order to achieve that, stemming, removal of closed class words, Word Sense Disambiguation (WSD) or Multiword Identification techniques can be applied combined or independently.
- **The comparison module** is based on two techniques that are explained later: Evaluating Responses with Bleu (ERB), which is more focused on the style of the answer; and, Latent Semantic Analysis (LSA), which is more focused on the content of the answer. Each of them gives as output a numerical score: ERB between 0 (bad answer) and 1 (good answer) and LSA between -1 (bad answer) and 1 (good answer). They can be used together or independently. In the case, that ERB and LSA scores are used together, it could be done as a linear combination of their normalized values in a common scale (e.g. from 0 up to

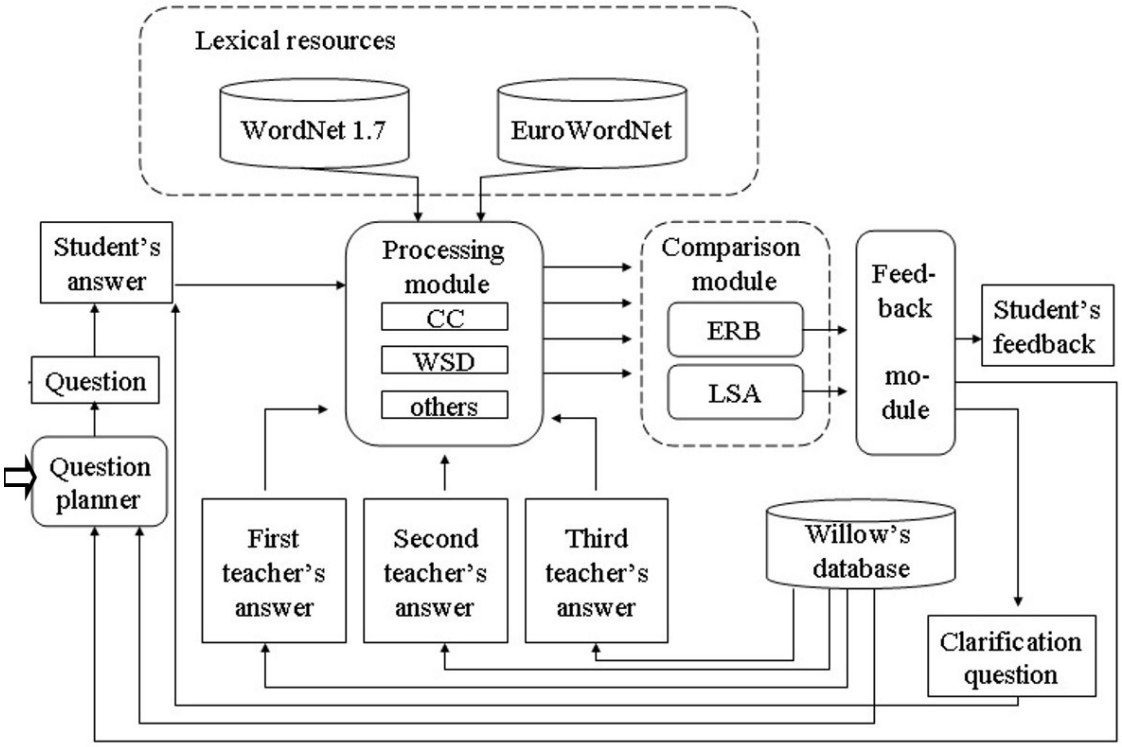


Figure 6.10: A simplified diagram of Willow's high-level architecture.

1).

- **The feedback module** that produces the final output to the student, that is, the numerical score, the processed answer and the references.

The high-level modules worked combined as follows: first of all, the system chooses a suitable question according to the level of difficulty that is able to manage the student in the related topic (e.g. easy, medium or high), language (e.g. Spanish or English), level of experience (e.g., novice or advance) and group of age (child or adult). Next, it waits for the student's answer. This answer and the references of the question chosen are processed with the selected wraetlic tools in Willoc. For instance, if stemming, removal of closed-class words and WSD have been chosen, the text will be stemmed, the closed-class words will be removed and the sense of each word will be identified. The processed texts enter the comparison module that applies the ERB and/or LSA technique to them. In this way, the feedback is generated.

However, whenever students fail a question, instead of showing them the feedback directly, they are forced to think more. In particular, students are presented the clarification dialogue, in which Willow starts asking them up to three questions from a more general to a more specific level of detail to guide them to the correct answer. Moreover, if a question is not passed the first time it is shown, the next time the processed answer will also be shown to help the student to answer.

Students are asked another suitable question until the amount of time defined by them is completed or until they have passed a certain number of questions. It is important to notice that as students answer more and more questions, Willow keeps track of how they are using the

relevant terms extracted from the references to update their CV and generate their conceptual models. Besides, a complete log system registers the options selected during each session, the time devoted to each question and the number of times the system has presented the same question to a student until s/he has been able to pass it.

6.5 Low-level architecture

Once the high-level architecture has been explained, in this section the focus is on detailing the internal modules. In particular, the processing module is described in Section 6.5.1, the comparison module in Section 6.5.2 and the feedback module in Section 6.5.3.

6.5.1 Processing module

The student's answer and the references of the question are processed with a set of NLP tools to transform them into an internal format, which is the input of the comparison module. It is important to mention that, for the sake of speed, the references are processed only once. Afterwards, the processed references are retrieved from the database and only the student's answer is processed.

There are several NLP techniques that can be applied in the processing module, and it is the administrator of the course who has to choose which ones must be applied. S/he has to do that at the beginning of each course by using another system in the Will tools package: the configuration tool called Willoc.

Once a set of techniques have been selected, they work together as a pipeline in which the output of one technique is the input of the next one (techniques not selected are skipped). All the currently available techniques are for the Spanish and English languages as included in the wraetlic tools and, they are the following ones [Alfonseca et al., 2006]:

- **Transformation to wraetlic**: Texts have to be translated to the wraetlic XML format to be processed by the wraetlic toolkit (see Appendix A for a sample of this format). The texts are broken into tokens (e.g. words, numbers and punctuation symbols) and the sentences boundaries are identified.
- **Stemming**: The words found are replaced by their canonical form (e.g. for verbs, the infinitive; for nouns and adjectives, the masculine singular). Stemming is an important step for free-text assessing as it allows the matching of words that could be inflected in different ways (e.g. *manage* and *managing*).
- **Removal of closed-class words**: Closed-class words are prepositions, conjunctions, etc. They could be useful for finding matching n-grams (block of n consecutive words). However, due to their lack of lexical meaning and their high frequency in every kind of text, they are sometimes irrelevant and, the wraetlic toolkit provides the possibility of removing them.
- **Word Sense Disambiguation (WSD)**: It is a classical problem in NLP as polysemous words have different senses. The sense intended by both the teacher and the student

should be identified to find out whether it is the same or not. In this way, the evaluation should be more accurate. In order to remove the ambiguity, each word w_i in the candidate and the references is replaced by the identifier of the synset in WordNet (e.g. collection is replaced by n06496793) as described in Ruiz-Casado et al. [2005].

- **Multiword term identification**: Sometimes, a group of words should be considered together. For instance, *Operating System* is the term, instead of *Operating* and *System* separately. In order to achieve this, the term identification module is used to compare the relative frequency of multiword expressions in a general corpus (e.g. the British National Corpus) with the frequency in the students' answers. Moreover, some filters can be applied to remove patterns that include verbs inside the multiword term.

Figure 6.11 shows how a sample student's answer is modified by each one of the techniques previously mentioned and some combinations of techniques. Other techniques that have been tried, although they have been finally discarded, as they have not improved the automatic free-text scoring accuracy are: syntactic dependence retrieval and translation into a logic formalism, and Anaphora Resolution to identify the referential expressions in the texts using the same techniques described for generating new references explained in Section 6.3.

6.5.2 Comparison module

Once the student's answer and the references have been processed with the selected NLP techniques, they enter the comparison module that gives as output, the feedback to the student (e.g. his or her numerical score and/or processed text). As stated before, this module is based on the ERB and LSA techniques that can be used independently or together as it is described below.

ERB

"Evaluating Responses with Bleu" (ERB) is a new statistical algorithm inspired in the BiLingual Evaluation Understudy (Bleu) algorithm [Papineni et al., 2001] [http24]. Bleu was created as a rapid procedure for evaluating and ranking Machine Translation (MT) systems. Bleu is based on an n-gram co-occurrence scoring procedure that has also been successfully employed to evaluate summarization systems [Lin and Hovy, 2003] or to recognize textual entailment [Pérez and Alfonseca, 2005e, Pérez and Alfonseca, 2006e].

The core idea of Bleu is that a system translation is better when it is closer to a set of human expert translations. In fact, its robustness stems from the fact that it works with several references against which it compares the candidate text. The use of several references, made by different human translators, increases the probability that the particular words and their relative order, in the automatic translation, will appear in some reference. On the other hand, the procedure is very sensitive to the choice of the references.

It is important to notice the similarity between the core idea of Willow and the original core idea of Bleu: the student's answer (the candidate translation in the original use of Bleu) is better when it is closer to a set of teachers' answers (the manual translations in the original use

Original: Collection of programs that supervises the execution of other programs and the management of computer resources. An operating system provides an orderly input/output environment between the computer and its peripheral devices. It enables user-written programs to execute safely. An operating system standardizes the use of computer resources for the programs running under it.
Stemmed: [Collection, of, program, that, supervise, the, execution, of, other, program, and, the, management, of, computer, resource, An, operating, system, provide, an, orderly, input, environment, between, the, computer, and, its, peripheral, device, It, enable, user-written, program, to, execute, safely, An, operating, system, standardize, the, use, of, computer, resource, for, the, program, run, under, it]
Without closed-class words: [Collection, programs, supervises, execution, other, programs, management, computer, resources, operating, system, provides, orderly, input/output, environment, computer, peripheral, devices, enables, user-written, programs, execute, safely, operating, system, standardizes, use, computer, resources, programs, running]
Stemmed, no closed-class words: [Collection, program, supervise, execution, other, program, management, computer, resource, operating, system, provide, orderly, input, environment, computer, peripheral, device, enable, user-written, program, execute, safely, operating, system, standardize, use, computer, resource, program, run]
WSD: [n06496793, of, n04952505, that, v01821686, the, n00842332, of, a02009316, n04952505, and, the, n00822479, of, n02625941, n11022817, An, operating, n03740670, v01736543, an, a01621495, n05924653, n11511873, between, the, n02625941, and, its, a00326901, n02712917, It, v00383376, user-written, n04952505, to, v01909959, r00152042, An, operating, n03740670, v00350806, the, n00682897, of, n02625941, n11022817, for, the, n04952505, v01433239, under, it]
WSD, no closed-class words: [n06496793, n04952505, v01821686, n00842332, a02009316, n04952505, n00822479, n02625941, n11022817, operating, n03740670, v01736543, a01621495, n05924653, n11511873, n02625941, a00326901, n02712917, v00383376, user-written, n04952505, v01909959, r00152042, operating, n03740670, v00350806, n00682897, n02625941, n11022817,n04952505,v01433239]
All synsets: [[Collection], [of], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [that], [v01615271, v01821686], [the], [n00068488, n00817656, n00842332, n11140581], [of], [a02009316], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [and], [the], [n00822479, n06765853], [of], [n02625941, n07941303], [n04334536, n04749592, n11022817], ...]
All synsets, no closed-class words: [[Collection], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [v01615271, v01821686], [n00068488, n00817656, n00842332, n11140581], [a02009316], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [n00822479, n06765853], [n02625941, n07941303], [n04334536, n04749592, n11022817], ...]

Figure 6.11: Modification of a student's answer depending on the configuration of Willow. The synset identifiers in the last four cases are taken from WordNet 1.7.

of BLEU).

Hence, BLEU needs a corpus of human references and a numerical similarity metric between the texts to compare irrespectively of the application. Papineni et al. [2001] based this similarity metric on the use of the Modified Unified Precision (MUP). The algorithm to calculate MUP is the following (let max(x) be the maximum of the set of values stored in x and $min(x, y)$ the minimum between the values a and b):

1. $lengthCandidate$ = { the number of words in the candidate text }
2. For each n-gram p in the candidate answer:
   (a) $refCount_r$ = {the frequency of p in each reference r }
   (b) $maxRefCount = max(refCount)$
   (c) $count$ = { the frequency of p in the candidate text }
   (d) $count^p_{clipped} = min(count, maxRefCount)$
3. $$sum = \sum_{p \in candidate} count^p_{clipped} \tag{6.1}$$
4. $$MUP = sum/lengthCandidate \tag{6.2}$$

This process has to be repeated for each value of n, usually for n from 1 to 4 (N=4), since for higher values of n, there will be very few occurrences of p in the references. According to their authors, MUP measures the translation adequacy for lower values of n and the translation fluency for higher values of n. In both cases, a higher MUP means a higher adequacy or fluency.

The next step in the computation of the similarity distance metric consists in combining all the values of MUP, for each n, in a single result. Papineni et al. [2001] concluded that the best results were attained when a weighted sum of the logarithms of MUPs was performed as shown in Equation 6.3.

$$combinedMUP = \sum_{i=0}^{MAX_N} \frac{log(MUP(n))}{MAX_N} \tag{6.3}$$

The last step is to modify the result of Equation 6.3 in order to penalize very short candidate texts, which might be incomplete. Besides, it should be kept in mind that a candidate is considered better when it is closer to the references and, thus, it should be expected that their lengths are similar. Only using $combinedMUP$, very short students' answers could achieve very good results in the case that the few words they have are all in the references. To correct this situation, Papineni et al. [2001] multiply the score given by Equation 6.3 by a factor, called the Brevity Penalty (BP) factor, calculated in the following way:

1. Find the reference translation whose length (measured in number of words) is the one most similar to the length of the candidate translation.
2. Let c be the length of the candidate text, and r be the length of the reference chosen in the previous step.
3. BP will be calculated as:

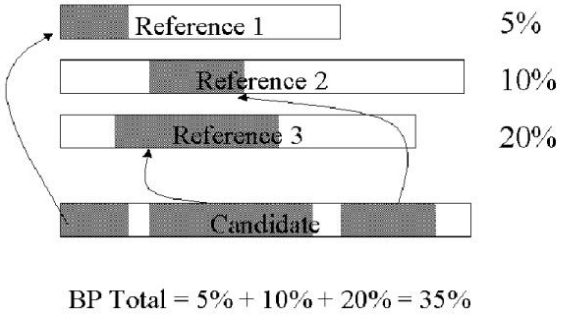


Figure 6.12: Procedure for calculating the Modified Brevity Penalty factor.

$$BP = \begin{cases} 1 & \text{if } c < r \\ e^{1-\frac{r}{c}} & \text{otherwise} \end{cases} \tag{6.4}$$

It can be observed that the penalty is not applied at sentence level, but to the complete text. Finally, the BLEU score of a student's answer (candidate) a is the result of Equation 6.5.

$$BLEU_{score}(a) = BP(a) \times e^{combinedMUP} \tag{6.5}$$

ERB follows the same steps that BLEU, but it has been slightly modified so that it takes into account not only the precision but also the recall. That is, to consider the percentage of the references that is covered by the student's answer. Hence, the BP factor has been modified into a new Modified Brevity Factor (MBP), which is calculated following this algorithm (see Figure 6.12 for a graphical description of the algorithm):

1. Order the references in order of similitude to the candidate text.
2. For n from a maximum value (e.g. 10) down to 1, repeat:
   (a) For each n-gram from the candidate text that has not yet been found in any reference,
   (b) if it appears in any reference, mark the words from the n-gram as found, both in the candidate and the reference.
3. For each reference text, count the number of words that are marked, and calculate the percentage of the reference that has been found.
4. The Modified Brevity Penalty factor is the sum of all the percentage values.

In this way, the ERB score of a student's answer a can be calculated according to Equation 6.6.

$$ERB_{score}(a) = MBP(a) \times e^{combinedMUP} \tag{6.6}$$

Finally, the output score of Equation 6.6, is between 0 and 1. However, teachers may need the score in other scale. Thus, in order to provide the student's score, the score has to be transformed to the teacher's scale. In order to achieve this, the following methods are proposed:

- If a set of student answers of previous courses marked by a teacher are available, then the

regression line per each question can be calculated.

- When a set of answers manually scored is not available, the regression line cannot be calculated, but estimated. This is done by assuming that the minimum score given by ERB (s_{min}) is 0 in the teacher's scale and the maximum score given by ERB (s_{max}) is 1 in the teacher's scale. Thus, the estimated regression line will be the line that crosses the points $(0, s_{min})$ and $(1, s_{max})$. However, it has the undesired consequence that in the case that a student achieves a score higher than the current s_{max}, the line is recalculated and some students' scores may be lowered down.

To sum up, the general procedure of ERB is as follows:

1. Count how many n-grams from the candidate text appear in any of the reference text (for n from 3 down to 0).
2. Clip the frequency of each n-gram with the maximum frequency with which it appears in any reference.
3. Combine the marks obtained for each value of n, as a weighted linear average.
4. Apply the MBP factor to penalize short candidate texts and take into account recall.
5. Calculate or estimate the regression line to translate the ERB's score to the teacher's scale.

The main advantages of ERB are that:

1. **It is very simple** to implement.
2. **It needs few lexical resources** just a set of references written in plain text without the need of big corpora that has to be trained or templates that have to be designed and filled.
3. **It is language-independent**, as the only compulsory processing done to the text is tokenization.
4. **It can be easily extended and improved**.

On the other hand, its main disadvantages are:

1. **It is very dependent on the choice of the references**, there should be several different references per question and they should be written trying to cover all the points that are expected from the students' answers.
2. **It is not suitable for all type of questions**, for instance creative questions, in which it is difficult to produce a correct answer, cannot be evaluated with ERB.

LSA

As it has been seen in Section 4.1, LSA [Deerwester et al., 1990, Foltz et al., 1998] is an unsupervised technique to estimate term and document similarity in a "cognitive" Latent Semantic Space. The LSA space is obtained by performing a Singular Value Decomposition (SVD) of the terms-by-documents matrix extracted from a large corpus. In other words, term co-occurrences in the corpus are captured by means of a dimensionality reduction operated on the terms-by-documents matrix. The vectors in the original space are mapped into a lower dimensional space,

Word	Medicine	Computer Science
HIV	1	0
AIDS	1	0
virus	0.5	0.5
laptop	0	1

Table 6.4: Example of Domain Matrix.

in which the similarity estimation is more accurate. The resulting LSA vectors can be exploited to estimate both term and document similarity.

LSA has already been used for assessing students' answers [Foltz et al., 1999, Dessus et al., 2000] with satisfactory results. Many different implementations of LSA can be found in the literature. For Willow, the ITC-irst LSA implementation has been used. In order to understand its particular details, it is necessary to review some concepts.

A LSA Domain Model is composed by soft clusters of terms. Each cluster represents a semantic domain [Gliozzo et al., 2004], i.e. a set of terms that often co-occur in texts having similar topics. A LSA Domain Model is represented by a $k \times k'$ rectangular matrix $\mathbf{D}$, containing the degree of association among terms and domains, as illustrated in Table 6.4.

LSA Domain Models can be used to describe lexical ambiguity and variability. Lexical ambiguity is represented by associating one term to more than one domain, while variability is represented by associating different terms to the same domain. For example the term `virus` is associated to both the Computer Science and Medicine domains (ambiguity) while the domain Medicine is associated to both the terms `AIDS` and `HIV` (variability).

More formally, let $\mathcal{D} = \{D_1, D_2, ..., D_{k'}\}$ be a set of domains, such that $k' \ll k$. A LSA Domain Model is fully defined by a $k \times k'$ *domain matrix* $\mathbf{D}$ representing in each cell $\mathbf{d_{i,z}}$ the *domain relevance* of term w_i with respect to the domain D_z. The domain matrix $\mathbf{D}$ is used to define a function $\mathcal{D} : \mathbf{R}^k \rightarrow \mathbf{R}^{k'}$, that maps the vectors $\vec{t_j}$, expressed into the classical Vector Space Model (VSM), into the vectors $\vec{t'_j}$ in the VSM domain. $\mathcal{D}$ is defined by

$$\mathcal{D}(\vec{t_j}) = \vec{t_j}(\mathbf{I^{IDF}D}) = \vec{t'_j} \tag{6.7}$$

where $\mathbf{I^{IDF}}$ is a diagonal matrix such that $i^{IDF}_{i,i} = IDF(w_i)$, $\vec{t_j}$ is represented as a row vector, and $IDF(w_i)$ is the *Inverse Document Frequency* of w_i.

Vectors in the VSM domain are called Domain Vectors. Domain Vectors for texts are estimated by exploiting Equation 6.7, while the Domain Vector $\vec{w'_i}$, corresponding to the word $w_i \in V$, is the i^{th} row of the domain matrix $\mathbf{D}$. In order to be a valid domain matrix, such vectors should be normalized (i.e. $\langle \vec{w'_i}, \vec{w'_i} \rangle = 1$).

In the VSM domain, the similarity among Domain Vectors is estimated by taking into account second order relations among terms. For example the similarity of the two sentences "*He is affected by AIDS*" and "*HIV is a virus*" is very high, because the terms `AIDS`, `HIV` and `virus` are highly associated to the domain Medicine.

Gliozzo et al. [2005] proposed the induction of LSA Domain Models from corpora using a

variation of LSA that is performed by means of a SVD of the term-by-document matrix $\mathbf{T}$ describing the corpus. The SVD algorithm can be exploited to acquire a domain matrix $\mathbf{D}$ from a large corpus $\mathcal{T}$ in a totally unsupervised way.

SVD decomposes the term-by-document matrix $\mathbf{T}$ into three matrixes $\mathbf{T} \simeq \mathbf{V}\mathbf{\Sigma}_{\mathbf{k}'}\mathbf{U}^T$ where $\mathbf{\Sigma}_{\mathbf{k}'}$ is the diagonal $k \times k$ matrix containing the highest $k' \ll k$ eigenvalues of $\mathbf{T}$, and all the remaining elements set to 0. The parameter k' is the dimensionality of the Domain VSM and can be fixed in advance[2]. Under this setting, the domain matrix $\mathbf{D_{LSA}}$ is defined as:

$$\mathbf{D_{LSA}} = \mathbf{I^N}\mathbf{V}\sqrt{\mathbf{\Sigma}_{\mathbf{k}'}} \tag{6.8}$$

where $\mathbf{I^N}$ is a diagonal matrix such that $\mathbf{i}^{\mathbf{N}}_{\mathbf{i,i}} = \frac{1}{\sqrt{\langle \vec{w_i'}, \vec{w_i'} \rangle}}$, $\vec{w_i'}$ is the i^{th} row of the matrix $\mathbf{V}\sqrt{\mathbf{\Sigma}_{\mathbf{k}'}}$.

Regarding document similarity, a variation of the *pseudo-document* methodology described in Berry [1992] is used, in which each document is represented by the sum of the normalized LSA vectors for all the terms contained in it, according to the *tf-idf* weighting schema commonly used in Information Retrieval and Text Categorization [Sebastiani, 2002].

According to Deerwester et al. [1990], in the LSA space, both *polysemy* (i.e. the ambiguity of a term that can refer to different concepts) and *synonymy* (i.e. the fact that the same concept, in a context, can be referred to by different terms) are implicitly represented. It is very important to consider those aspects when evaluating students' answers. For example both *pc* and *laptop* can be used to denote a computer; *architecture* has a sense in the field COMPUTER_SCIENCE and a different one in the field BUILDING_INDUSTRY.

Polysemy and synonymy are modeled by exploiting the information from an external corpus, providing the system of an "a-priori" semantic knowledge about the language, represented by a structure of semantically related terms. Such structure allows the system "to see" more than the content actually expressed by the words themselves, improving the superficial text comprehension obtained by a simpler string matching.

To sum up, the LSA algorithm, used to evaluate the students' answers, is defined as follows: let $\vec{a}$ be the pseudo-document vector obtained from the student's answer a and let $R = \{\vec{r_1}, \vec{r_2}, \ldots, \vec{r_n}\}$ be the set of the pseudo-document vectors corresponding to the references; the LSA score is defined by the mean of the pseudo-document similarities between $\vec{a}$ and each vector $\vec{r_i} \in R$. This score is then normalized in order to return a value in the range [0,1], as defined by Equation 6.9.

$$LSA_{score}(a) = \frac{\sum_{\vec{r_i} \in R} \cos(\vec{a}, \vec{r_i})}{2|R|} + 0.5 \tag{6.9}$$

The same procedures described in Section 6.5.2 for scaling this score to the teachers' scale are applicable to the score produced for Equation 6.9. The main advantages of using LSA are:

- **It is oriented to measuring semantic content**, using LSA, two words are related not because they are one next to the other, but because one can be replaced in the context of

[2] 400 dimensions have been used as suggested by Alfio Gliozzo, personal communication, ITC-irst, 2005.

the other without changing the meaning of the sentence.

- **It can deal with lexical ambiguity and variability**, as explained above.
- **It can deal with polysemy and synonymy**, as explained above.

On the other hand, the main disadvantages are:

- **It requires an initial training step**, in order to tune some values of the LSA parameters.
- **It is a bag-of-words technique**, that is, the order of the words is not taken into account and the system can be easily fooled.
- **It requires complex lexical resources**, it is necessary big corpora to do the initial training step.

ERB+LSA

The LSA and the ERB algorithms differ substantially with respect to the type of linguistic analysis performed. In addition, LSA accesses an external knowledge source. Hence, the assessments of ERB and LSA can be considered independent. Thus, it is expected that their individual performances will be increased by adopting a system combination schema [Florian et al., 2002]. The combination schema adopted is a weighted sum of their outputs, as described by Equation 6.10 for a student's answer a.

$$Willow_{score}(a) = ERB_{score}(a) \times \alpha + LSA_{score}(a) \times (1 - \alpha) \tag{6.10}$$

α is a parameter that allows us to assign in advance a weight to ERB or to LSA. In spite of its simplicity, this combination schema is effective and very general. When α is set to 0.5, equal weights are assigned to both systems. Several experiments have been tried to optimize α as shown in Section 8.4.

It is also important to notice that as ERB's score is between 0 and 1, and the LSA score has been normalized to be also between 0 and 1, so that both are in the same scale and can be combined. Thus, the resulting score will also be in the [0-1] scale and the same procedures described in Section 6.5.2 for scaling this score to the teachers' scale can be applied.

Finally, it is interesting to observe how using ERB and LSA together allows us to combine the advantages of both techniques, while overcoming some of their disadvantages. In particular, the bag-of-words problem of LSA is overcome using ERB, which is extended so that the comparison is not only focused on lexical information but can also deal with semantic information.

6.5.3 Feedback module

The feedback items that Willow provides to the instructor and to the student are different. For the instructors, it is the conceptual model of each individual student and the whole class as will be shown in Chapter 7. That way, they can see how the concepts exposed in the lessons are being understood and act accordingly (see Section 5.4.6).

While for the students (due to Willow's main goal of formative assessment) it may consist of:

- **The numerical score** as the result of the comparison module. Its main goal is to provide the student with an orientation about how well s/he has answered according to the numerical scale provided by the teacher. It would be useful, for instance, when students have to pass a certain cutoff score (e.g. 5 in a [0-10] scale), they can get more training until they consistently get a score around the cutoff aimed.
- **The processed answer**, that is a copy of the student's answer with the portions that are correct (as they are more similar to the references) marked in green.
- **His or her conceptual model and the whole class conceptual model** as will be explained in Chapter 7.
- **Indications of progress** in their assessment session: the level in which they are, the number of questions remaining and/or the remaining time. This way, they can get a better organization of the time devoted to the study of the course.
- **Previous feedback** of the questions that have previously failed. In particular, the score they got and their processed answer. That way, they are reminded which portions were not marked in green and, they can focus on fixing the missed points.

Anyway, students can choose which items they want to be given as feedback. This is because it has been proved that, despite the many benefits feedback has, too much feedback could also be disadvantageous [Lilley et al., 2005].

6.6 Optimum use of Willow

It is important to highlight that unlike other free-text CAA systems, Willow does not need any training with previous courses' answers. Nevertheless, in order to reach its optimum use, and provided that for each question, there are a small set of references written by the teachers and a big set of the best student's answers (i.e. scored with the maximum grade in a course), it is possible to choose the references for the next course according to the following procedure:

1. `Join the set with the teacher's references and the student's answers.`
2. `Divide it randomly into two subsets. One subset will be taken as references.`
3. `Score the answers in the other subset against those references to measure the quality of that set of references.`
4. `Repeat 2 and 3 steps for each possible subset choice.`

However, as can be calculated, just with a simple set of 100 texts, the number of possible references is the size of the superset (2^{100}) and it would be computationally very expensive to repeat step 2 as many times. This is the reason why a genetic algorithm has been chosen to guide the search. A genetic algorithm [Holland, 1992] is a type of evolutionary algorithm. It is inspired by biology and thus, based on techniques such as inheritance, mutation, selection and crossover to search a solution for a certain problem. Solutions are usually represented in binary as strings of 0s and 1s.

First of all, in order to use a genetic algorithm, an initial population of abstract representations (i.e. chromosomes or genotype of the genome) of candidate solutions (i.e. individuals, creatures or phenotypes) have to be generated. Next, the algorithm iterates from this popula-

tion, calculating the fitness of each individual and selecting which ones should be combined to create new possibilities for the next generation (as they have a high fitness value) and which ones should be discarded (as they have a low fitness value). The process stops after a number of generations or when a certain threshold is fulfilled.

In Willed, the initial population is represented as a binary string with as many digits as the total number of best students' answers and references. The students' answers are represented by a 0 (as they are not initially considered to be used as references this new course) and the references are represented by a 1 (as they are considered to continue being used as references this course). This population evolves using the genetic algorithm implemented in the PGAPack [Levine, 1996] [http25]. The stop condition has been fixed as one hundred iterations without improvement of the fitness function that is the Pearson correlation between the automatic and teachers' scores using this set of references. Finally, the solution achieved is parsed so that the new 1s in the string determine which are the new references to be used in the course. Please, notice that some best students' answers can become references this course and some texts that were references the previous course could not be used again.

Moreover, it is possible to calibrate some internal parameters for the next course's training. In particular, to tune the values of the regression line to produce a more accurate numerical score based on the teachers' scores. Hence, for a set of questions, Willow's scores are stored in y. Next, when a teacher selects the calibration option in Willed, s/he is asked to provide a score to a set of questions previously marked by Willow (the automatic scores are not shown to the teachers to avoid any kind of bias). In the case that several teachers evaluate the same question, as the aim is not to measure the human interrater agreement, the last score provided is the one stored. The teachers' scores are stored in x. Let $cov(x, y)$ be the covariance of x and y, and $var(x)$ the variance of x, then the calibration process finds the value of b according to Equation 6.11.

$$b = \frac{cov(x, y)}{var(x)} \tag{6.11}$$

$$y = \bar{y} + b \times (x - \bar{x}) \tag{6.12}$$

Thus, whenever Willow detects that a question has stored values for $\bar{x}, \bar{y}$ and b different than zero, then it uses them to calculate the automatic score as given by Equation 6.12.

Finally, to sum up, the optimum use of Willow can be reached from the information gathered from one course to the following one, by:

1. `Obtaining the set of answers from the students.`
2. `Marking them with Willow, and supervising the markings.`
3. `Putting together all the correct answers from the students and the original references.`
4. `Using genetic algorithms and, finding the reference set that gives the highest correlation between the human and automatic scores.`

5. Using anaphora resolution and, generate more references from the ones selected in the previous step.
6. Calibrating the values of $\bar{x}, \bar{y}$ and b to calculate the regression line and producing the scores in the scales indicated by the teachers.

Chapter 7

An example of conceptual viewer: COMOV

Once the conceptual model has been generated, it can be represented in several knowledge representation formats with a conceptual model viewer such as the one presented in this Section and implemented as one of the Will tools: COMOV, an on-line COnceptual MOdel Viewer with Natural Language Support. At present, five different representations have been included in COMOV to show the generated conceptual model: a concept map, a conceptual diagram, a table, a bar chart and, a textual summary.

In order to use COMOV, the teacher or the student has to be registered in the system. It is important to take into account, that students who have a user and a password to log into Willow can use the same user and password to log into COMOV. Besides, that teachers who have a user and a password to log into Willed can use the same user and password to log into COMOV.

That way, whenever the teacher or the student wants to see a conceptual model, they only have to access COMOV. For an individual's conceptual model, the student's identification number has to be introduced and, for the whole class, the field devoted to the identification number has to be left in blank (see Figure 7.1).

COMOV represents the conceptual model always updated with the information gathered from the students' answers. This permits to capture the conceptual evolution of the students since the conceptual models generated at different times can be stored and reviewed them later in sequence.

Table 7.1 gathers the information about COMOV in order to be easily compared to other systems that are underpinned by the use of conceptual models such as the ones presented in Chapter 3. It can be observed how COMOV extends many of the possibilities of the reviewed systems as it is not limited just to one format of knowledge representation or to only one language. On the other hand, COMOV is able to manage five different knowledge representations and displayed them in Spanish or in English.

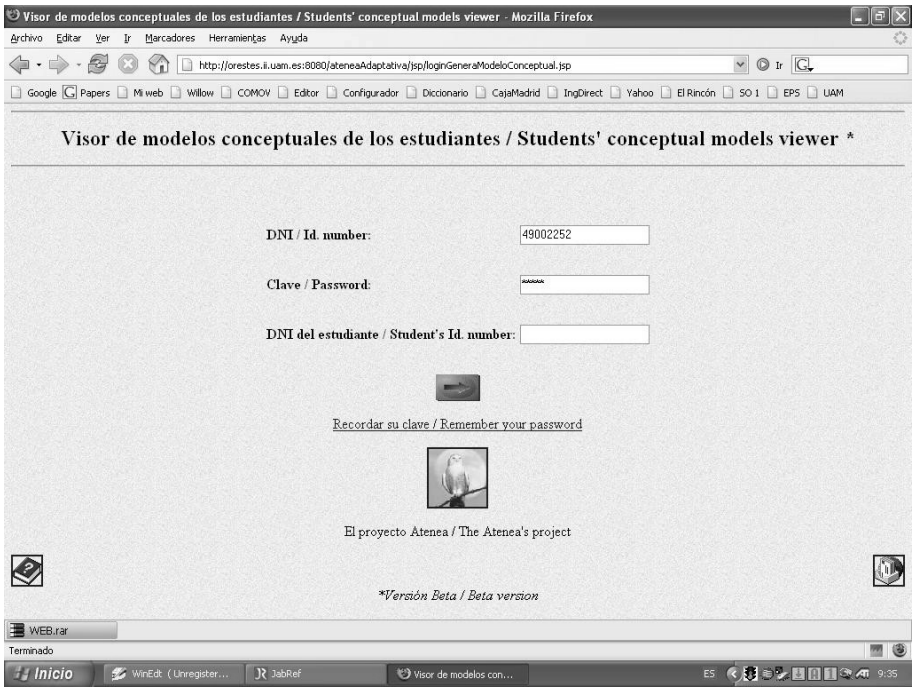


Figure 7.1: A snapshot of the COMOV's login page.

Name	**COMOV**
Goal	Show conceptual models
Domain	Computer Science
Language	Spanish+English
Type	Inspectable
Addressee	Student+Instructor
Knowledge Representation Format	Concept map, conceptual diagram, table, bar chart and textual summary
Technique	Free-text Adaptive Computer Assisted Assessment

Table 7.1: COMOV values for the second comparison table presented in Chapter 3.

7.1 Concept map

As it has previously seen, concept maps have been widely used for many aspects of education and they are considered as one of the best representations to intuitively show how concepts are interrelated in people's minds and, where the misconceptions and lack of previous concepts are. This has been the motivation to include this form of knowledge representation in COMOV.

A concept map is composed by nodes representing concepts, and links between them. A spider-like organization of the map has been chosen, as it is one of the most suitable formats for the hierarchy of concepts proposed.

Furthermore, given that the number of concepts of the conceptual model is higher than a certain threshold (e.g. 50), the CLOVER tool with clustering possibilities [Freire and Rodríguez, 2004] is used to visualize the conceptual model as a concept map. On the other hand, if the number of concepts is lower, and to have a concept map representation of the conceptual model simpler and more similar to Novak's concept maps, a new tool called IOV can be used. In any case, the goal is that by looking at the shape of the concept map, teachers can see how well

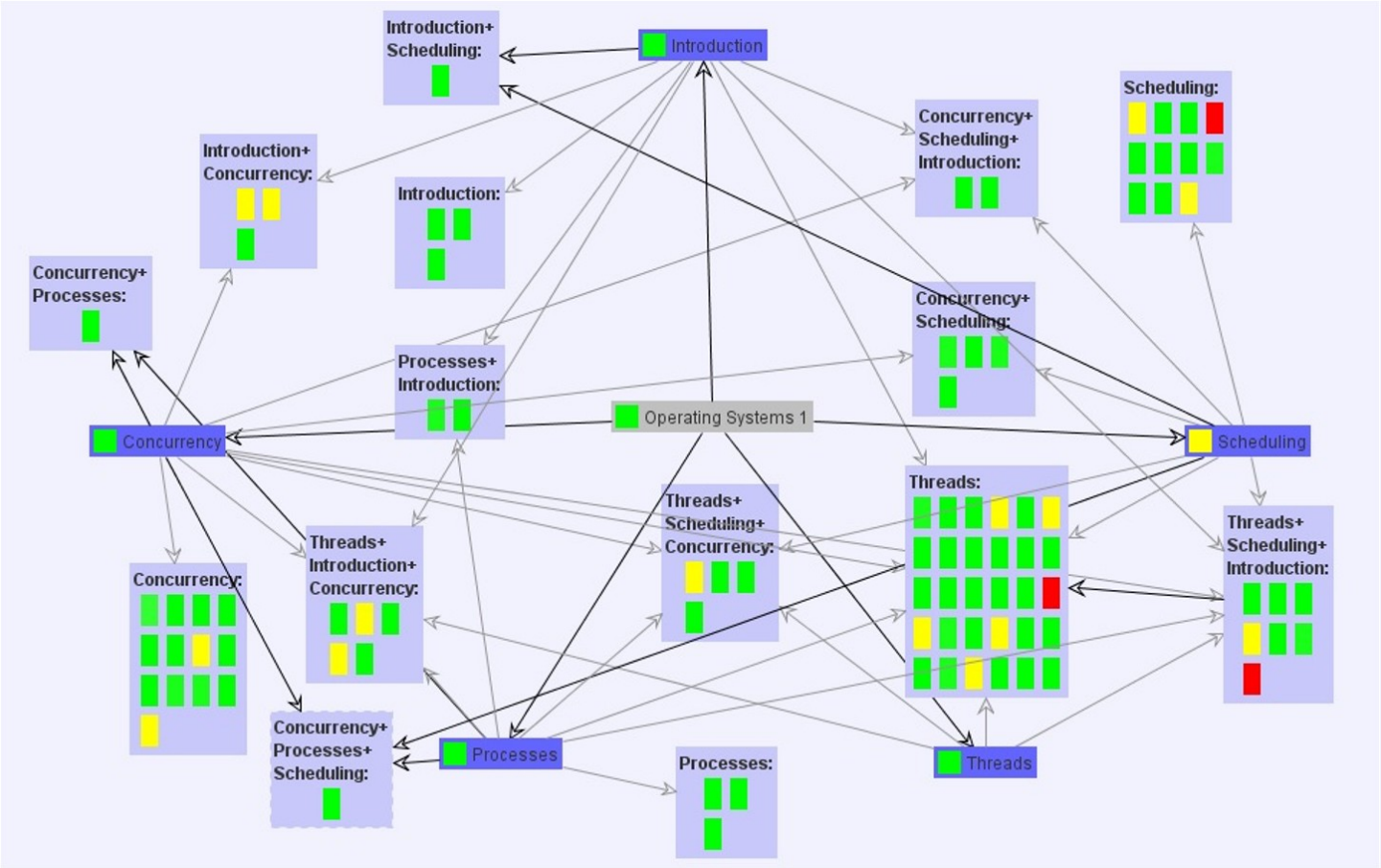


Figure 7.2: An example of a good student's conceptual model represented as a concept map using CLOVER.

students have assimilated the concepts exposed in the lessons.

Figures 7.2 and 7.3 show two examples of concept map displayed by CLOVER, for two different students, with high and low scores respectively. It can be seen that a background color schema was used to indicate the type of node: white for BCs, blue for TCs and grey for AC, while the foreground color represented the CV of each concept or group of concepts: red for unknown concepts with CV less than 0.4, yellow for uncertain concepts with CV between 0.4 and 0.6 and green for known concepts with CV above 0.6. Moreover, as CLOVER uses a customized clustering algorithm to aggregate related concepts together and reduce visual clutter, some BCs were automatically grouped according to the TCs they had been assigned to.

Figure 7.4 shows an example of a student's conceptual model represented as concept map with IOV. As can be seen, there have been several modifications: the clustering has been deactivated, i.e. a node in IOV always represents a concept; the color schema has changed so that there is no longer two different color codes, but only one that serves to indicate the CV of the concept and goes from utter red (CV=0) up to utter green (CV=1); the type of node is indicated by the size and place in the concept map: the AC is bigger and it is always at the center, the TCs are medium-size and are placed in the second radial line, while the BCs are

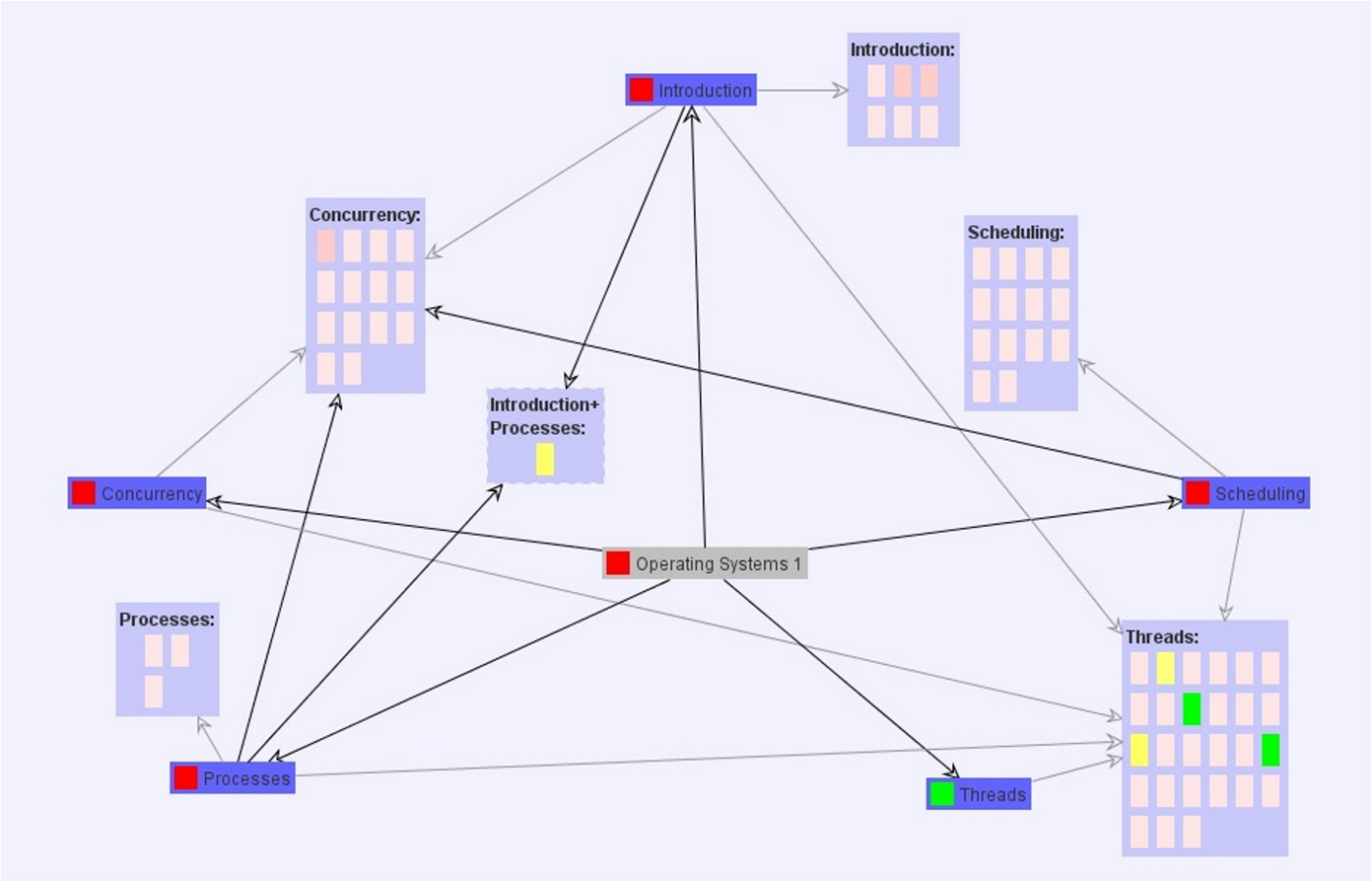


Figure 7.3: An example of a bad student's conceptual model represented as a concept map using CLOVER.

smaller and are placed in the next radials lines; and, the links have been reorganized in an effort to avoid crossings among links.

In this way, it should be easy to discern if the student has successfully assimilated the concepts exposed in the lesson just by looking at the higher concept of the hierarchy (the AC). If it has a green foreground color, it means that the student is ready to continue learning another AC. Otherwise, some problems have appeared and they can be identified by looking at lower concepts in the hierarchy, initially TCs to see which ones are lacking and next, the BCs related to the non green TCs.

Additionally to the information provided by the concepts and its hierarchy, links are very useful to detect misconceptions and lack of relationships. The misconceptions are detected whenever there is a type 3 link between two BCs that should not be related and thus, teachers should explain why. On the other hand, the lack of type 3 links denotes that students may understand each isolated concept but they have not recognized that they are related and thus, teachers need to reinforce the link between them.

This representation in form of concept map is particularly interesting whenever a global view of a particular student or the whole class is pursued. Moreover, when the goal is to follow

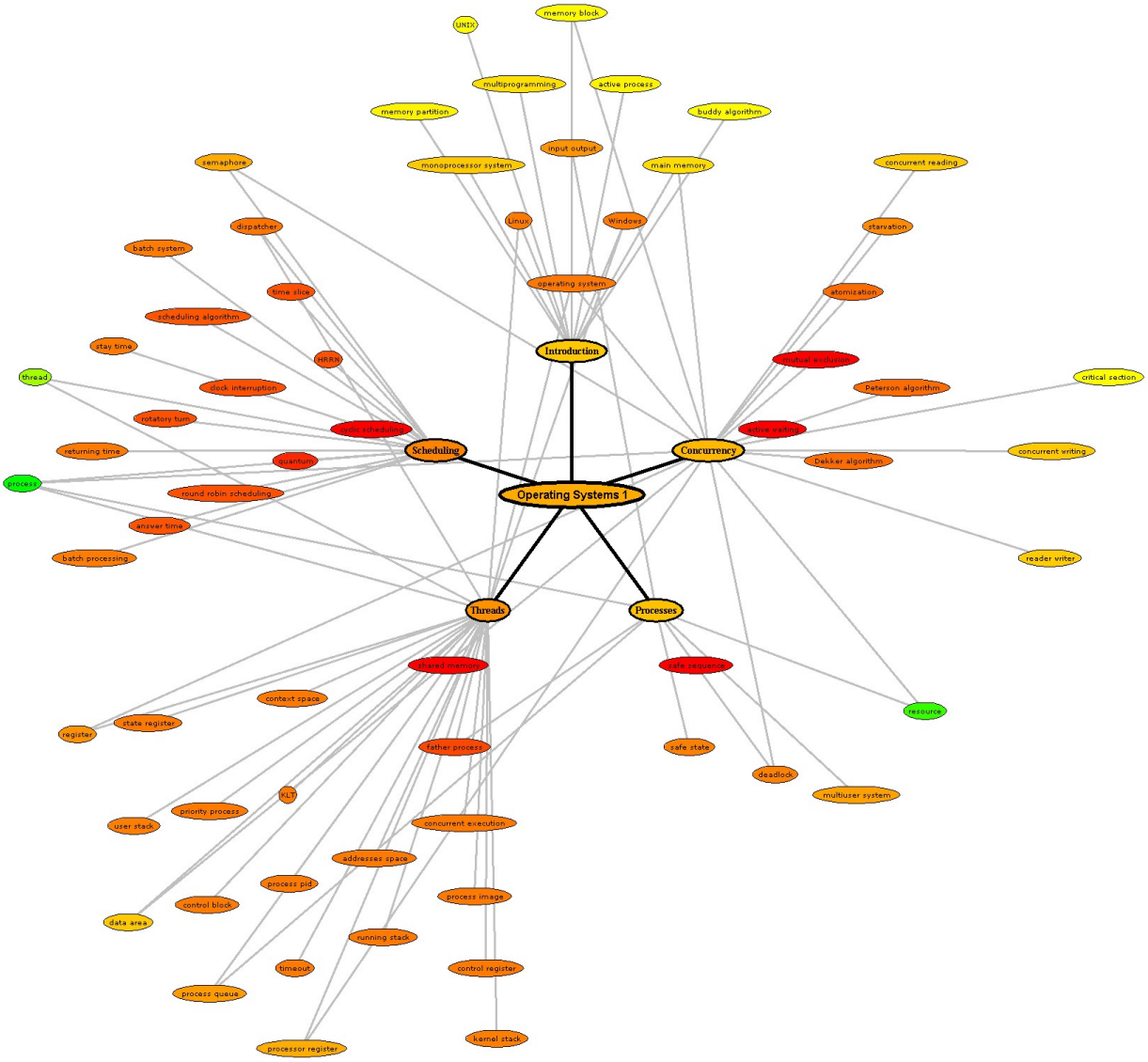


Figure 7.4: An example of concept map of a student using IOV.

the students' conceptual evolution (see Figures 7.5 and 7.6). The first picture captures the conceptual model of a group of students enrolled in an Operating Systems course just a few weeks after the beginning of the course. Thus, the structure is very simple: the AC in the center, the five TCs of the five lessons under study linked to it and the BCs automatically extracted from the references of the course. Only one topic has been learnt. Some months later, after students have been answering questions to the free-text ACAA system and more topics have been studied, it can be seen how lessons have been completed (almost all the concepts have a light background, green) and new links have been created between the concepts as they have been assimilated by the students and appeared in their answers.

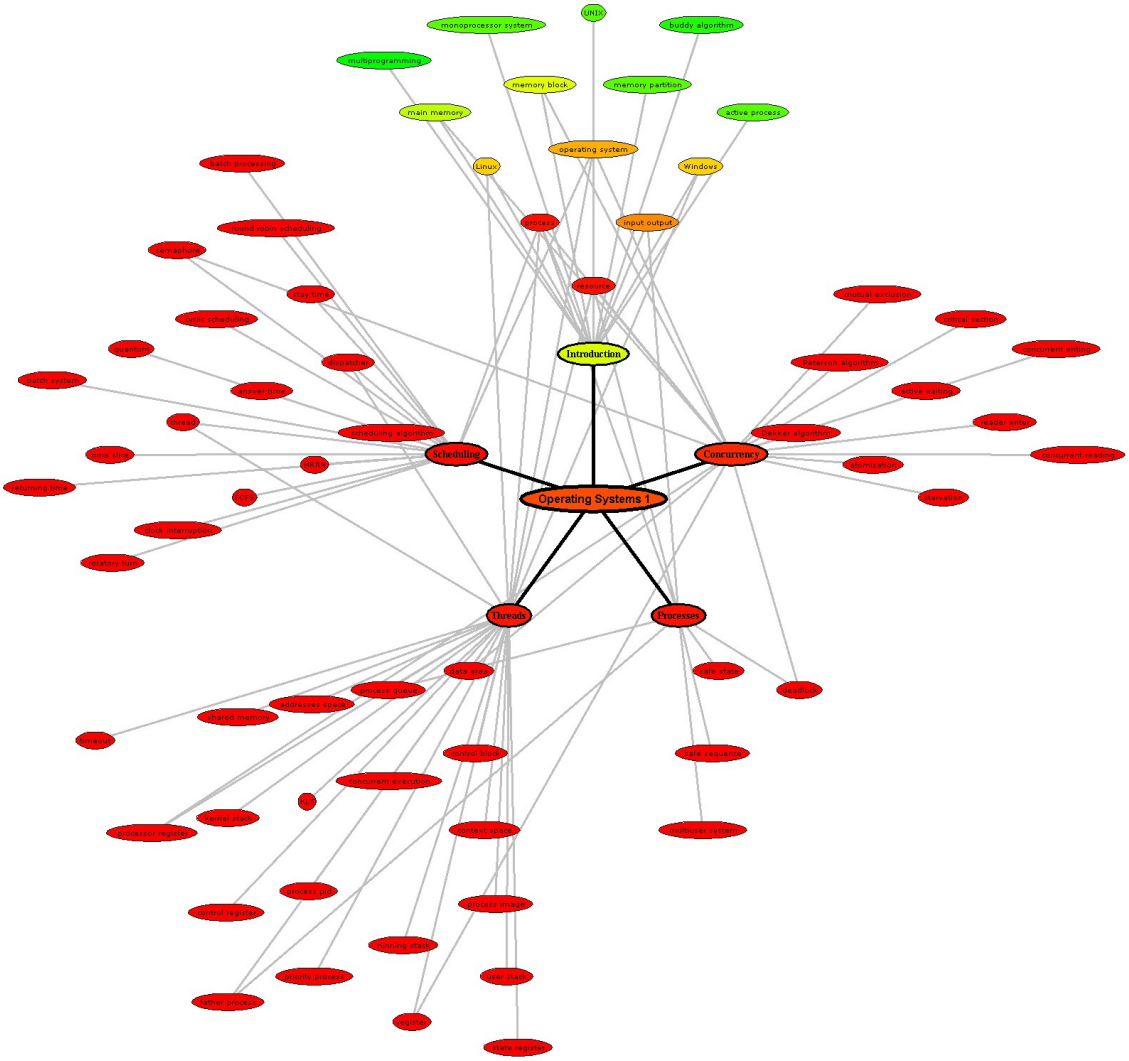


Figure 7.5: An example of concept map of a class that has started a course.

7.2 Conceptual diagram

The conceptual model represented as a concept map can be sometimes shown not in the traditional format of a concept map, but as a hierarchical diagram with the most important concept at the top and less relevant concepts below. This way, it is easier to check which concepts are more important and how they have been assimilated just by looking at their place in the diagram. See Figure 7.7 for a snapshot of a student's conceptual diagram representation.

The focus of the conceptual diagram is on the concepts. Thus, to make the representation simpler, the relationships among them are not explicitly represented. Besides, the concepts are organized upon three types of rows:

- **The AC row**: It is the first row placed at the top of the diagram. There is only one column because there is only one AC per conceptual model.
- **The TCs row**: The second row has as many columns as lessons included in the AC under study. The order in which the TCs are presented in this row is the same in which they

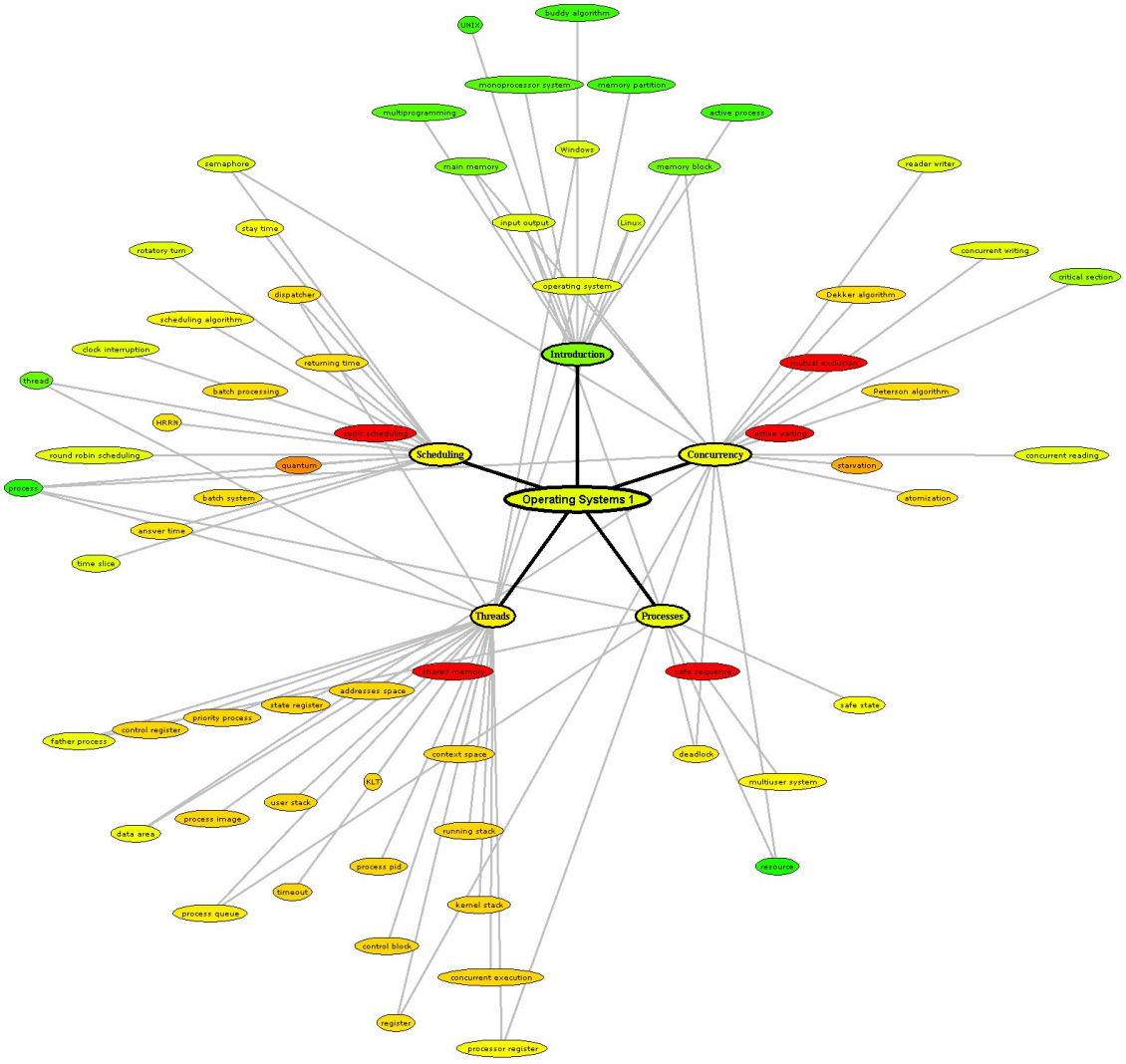


Figure 7.6: An example of the concept map of the same class but some months later.

have been stored by the teacher using the free-text ACAA authoring tool.

- **The BCs rows**: The rest of the rows correspond to the BCs related to the TCs, which are ordered from lower CV values (less known BCs) to higher values (best known BCs). There can be repetitions across TCs provided that the same BC belongs to two or more different TCs. It is also possible that the height of each column is different as the number of BCs related to each TC is different (automatically extracted from the term identification module from the references of each TC).

Hence, it can be seen that, unlike the concept map, the type of concept is no longer indicated by the size of the node, but by the place in the diagram: AC in the first row, TCs in the second row, and BCs from the third row to the end of the diagram.

Regarding the CV, the same background color schema of the concept map is used. That is, the CV of the concept determines the background color from red (CV=0) to green (CV=1) passing from light red, orange, yellow and light green. So, just by looking at the color of the

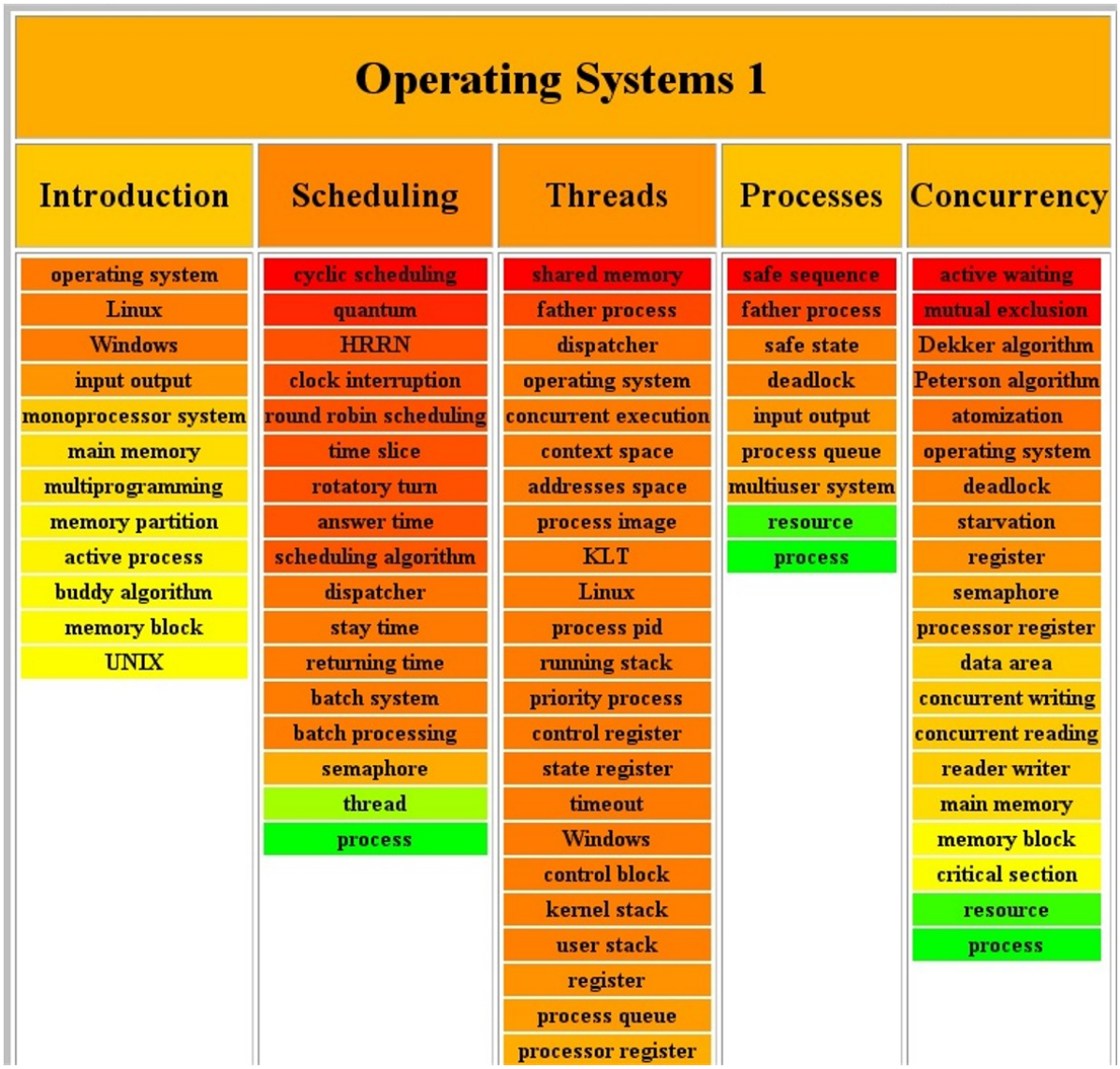


Figure 7.7: An example of conceptual model represented as a conceptual diagram.

top row, the diagram gives a clear indication of the general level of understanding of the topics of the area-of-knowledge under study. Additionally, for the users that would like to have the exact numerical value of the CV for each concept, a tooltip has been included so that when they pass the mouse over each cell, the numerical value of the CV is shown in the tooltip. Finally, the evolution of how a certain concept has been assimilated can be followed by looking at it in the diagram captured in different instants through a period of time. This can be done for one student or for the whole class.

7.3 Table

The goal of this format of representation of the students' conceptual model is to focus on the BCs and how well the students seem to have understood them by looking at the exact CV

assigned by the system. Thus, neither links nor TCs and AC are displayed. Moreover, if the table is ordered from higher to lower CV, teachers and students can easily see which concepts are better understood. Conversely, if the table is ordered from lower to higher CV, they can see which concepts have still not been assimilated and should be reviewed.

As can be seen in Figure 7.8, each row represents a BC, and there are five columns indicating for each BC its:

- **Term**: That is, the label of the BC as extracted by the automatic Term Identification module from the references.
- **Weight:** Each BC has a weight associated which indicates its relevancy. It is calculated from its frequency of appearance in the teachers' references. It has been normalized between 0 and 1 so that the more similar to 1, the most important the BC is as it is very common in the references.
- **ScoreConfidence**: The value of the first metric to calculate the CV (see Equations 5.13 and 5.20). As it has previously seen in Section 5.4.4, it is related to the score given by Willow to the answers in which the term appears. This intermediate value is shown to let the teacher distinguish how the CV has been calculated from the combination of this value and the value in the RateConfidence column. It is also normalized between 0 and 1, and the higher the better.
- **RateConfidence**: The value of the second metric (see Equations 5.14 and 5.23). As it has been explained in Section 5.4.4, it is based on the comparison of the frequency of use of the BC in the student's answer compared to the frequency of use of the BC in the references. It is also normalized between 0 and 1.
- **Confidence-value (CV)**: The combination of ScoreConfidence and RateConfidence according to Equations 5.15 and 5.21. It is the most important value in the row, as it is the value used by the free-text ACAA system to ascertain how well the student, or group of students, seems to understand the term. The nearest to 1, the better.

The table can be ordered according to several columns. Hence, for instance, by sorting by weight factor, teachers can see how well the students understand the most important terms and focus on them.

The same table is shown for the student and the class conceptual model. The difference is that the values for the student's conceptual model are particular to his or her answers while for the class conceptual models the values have been calculated from the answers of all the students and, thus, they guide the teacher to have a general overview of the class understanding of the concepts.

7.4 Bar chart

It seems natural that, once the information has been gathered as a table, it can also be visually represented as a chart. Several types have been considered, including columns, bars, linear, circular, bubbles, area or radial charts. Finally, the bar chart has been chosen because it is a simple yet powerful way of representing the BCs and their confidence value as relative

Term	Weight	ScoreConfidence	RateConfidence	Confidence Value (CV)
process	1.0	0.8	1.0	0.9
resource	0.9	1.0	0.3	0.6
thread	0.3	0.8	0.1	0.5
critical section	0.9	0.6	0.0	0.3
memory block	0.1	0.6	0.1	0.3
buddy algorithm	0.1	0.6	0.0	0.3
data area	0.3	0.5	0.1	0.3
main memory	0.9	0.5	0.1	0.3
reader writer	0.1	0.5	0.0	0.3
UNIX	0.9	0.6	0.1	0.3
active process	0.9	0.6	0.0	0.3
memory partition	0.9	0.6	0.0	0.3
multiprogramming	0.7	0.5	0.1	0.3
monoprocessor system	0.6	0.5	0.0	0.3
concurrent reading	0.5	0.5	0.0	0.3
concurrent writing	0.4	0.5	0.0	0.3
operating system	0.3	0.3	0.0	0.2
control block	0.3	0.3	0.0	0.2
deadlock	0.9	0.3	0.1	0.2
multiuser system	0.1	0.4	0.0	0.2
semaphore	0.1	0.4	0.0	0.2
starvation	0.9	0.4	0.0	0.2
input output	0.1	0.4	0.0	0.2
process queue	0.1	0.4	0.0	0.2
register	0.1	0.3	0.1	0.2
return time	0.1	0.3	0.0	0.2
stay time	0.1	0.3	0.0	0.2

Figure 7.8: An example of table ordered from higher to lower confidence-value basic-concepts of a student's conceptual model.

percentages.

As in the table, neither links nor TCs and AC are represented in order to let the user focus on the BCs and how well they have been assimilated. Hence, each BC is represented by a bar and the length of the bar indicates its CV. It is important to highlight that, in this case, not the exact value is given, but the relative percentage. Moreover, the color of the bar can be (see Figure 7.9):

- Blue, for BCs with CV lower than 40%.
- Yellow, for BCs with CV between 40% and 60%
- Green, for BCs with CV higher than 60%.

All these values are parameterizable and can be changed by the administrator. Besides, the order in which BCs are displayed can be chosen according to ascendent or descendent CV

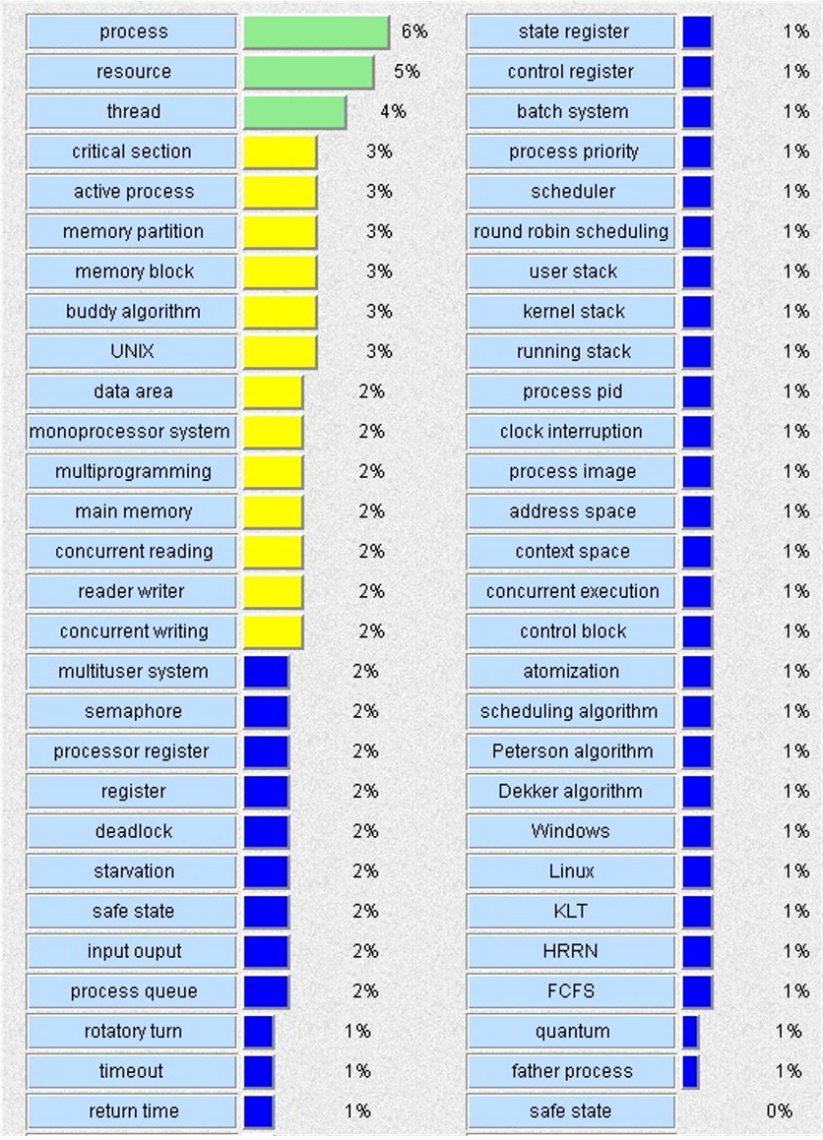


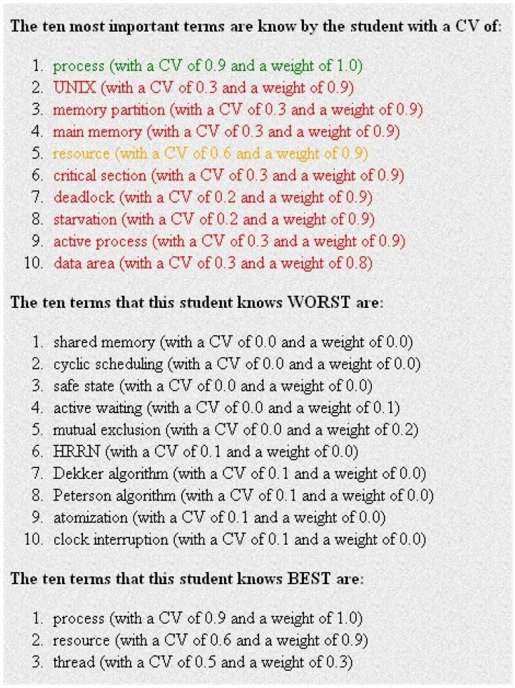
Figure 7.9: An example of bar chart ordered from lower to higher confidence-value basic-concepts of a student's conceptual model.

values.

Finally, for the class conceptual map, all it is the same, taking into account that it helps to see the level of assimilation of the BCs exposed in the lessons not only for one student but, for the whole class.

7.5 Textual summary

The system can also generate text summaries: one report per student and a class report. Similarly as in the table and the bar chart, text summaries focus on the BCs. Each report consists of three ordered lists in which each line gives information about a certain BC. In particular, it shows its name, CV and weight. The three ordered lists are as follows (see



The ten most important terms are know by the student with a CV of:

1. process (with a CV of 0.9 and a weight of 1.0)
2. UNIX (with a CV of 0.3 and a weight of 0.9)
3. memory partition (with a CV of 0.3 and a weight of 0.9)
4. main memory (with a CV of 0.3 and a weight of 0.9)
5. resource (with a CV of 0.6 and a weight of 0.9)
6. critical section (with a CV of 0.3 and a weight of 0.9)
7. deadlock (with a CV of 0.2 and a weight of 0.9)
8. starvation (with a CV of 0.2 and a weight of 0.9)
9. active process (with a CV of 0.3 and a weight of 0.9)
10. data area (with a CV of 0.3 and a weight of 0.8)

The ten terms that this student knows WORST are:

1. shared memory (with a CV of 0.0 and a weight of 0.0)
2. cyclic scheduling (with a CV of 0.0 and a weight of 0.0)
3. safe state (with a CV of 0.0 and a weight of 0.0)
4. active waiting (with a CV of 0.0 and a weight of 0.1)
5. mutual exclusion (with a CV of 0.0 and a weight of 0.2)
6. HRRN (with a CV of 0.1 and a weight of 0.0)
7. Dekker algorithm (with a CV of 0.1 and a weight of 0.0)
8. Peterson algorithm (with a CV of 0.1 and a weight of 0.0)
9. atomization (with a CV of 0.1 and a weight of 0.0)
10. clock interruption (with a CV of 0.1 and a weight of 0.0)

The ten terms that this student knows BEST are:

1. process (with a CV of 0.9 and a weight of 1.0)
2. resource (with a CV of 0.6 and a weight of 0.9)
3. thread (with a CV of 0.5 and a weight of 0.3)

Figure 7.10: An example of textual summary of a student's conceptual model.

Figure 7.10):

- **The first list identifies how well the ten most important BCs (i.e. with the highest weight) have been understood** by a student or a group of students. It is ordered from higher to lower CV and similarly as with the conceptual diagram, the use of the color schema is combined with the exact numerical value. In fact, the numerical value is written next to the term, and the font color of the text changes according to the confidence value:
  - Red, for CV lower than 0.4
  - Yellow, between 0.4 and 0.6
  - Green, higher than 0.6

  All these values are parameterizable and can be changed by the administrator. This list is helpful because it allows teachers and students to focus just on these ten most important BCs and avoid giving too much information that can be overwhelming in some cases (e.g. when the course is big with many TCs and BCs). Moreover, not all BCs have the same relevancy, and thus, it is more important that students know better a more relevant BC than a less relevant one.
- **The ten worst known concepts**: The second list contains the ten BCs that are worst understood by the student or the class as they have the lowest CV. A configurable threshold of 0.5 has been established so that, in this list, BCs with a CV higher than 0.5 cannot be included. It is possible that the list is not complete with ten concepts provided that there are no ten concepts with CVs lower than 0.5. It is even possible that the list would be empty when the course finishes and no BCs should remain unknown by the students

that have studied. This list is specially relevant, during the course, in the case of the class conceptual model as it evidences the necessity of stop teaching new concepts until the previous ones have been understood and, meaningfully linking to them can be done. It is also important to take into account the weight of each BC. It is because in the case that students do not understand low weight BCs (i.e. less important BCs) it is less problematic, than in the case that students do not understand high weight BCs (i.e. more important BCs).

- **The ten best known concepts**: The third list contains the ten BCs that are best understood by the student or the class as they have the highest CV. The same threshold that above has been used, so that BCs with CV lower than 0.5 are not included. Again, the list could be incomplete until ten BCs have a CV higher than 0.5. In fact, at the beginning of the course all the lists start empty as all BCs have a zero CV. During the course, as students start answering questions, this list starts to grow and it is expected that by the end of the course, can be completed with the most relevant BCs. It is also important to highlight that in the case of the class conceptual model, it gives the list of concepts that should not be reviewed again as they have already been correctly assimilated by the class.

7.6 Recap of the main points of the five representation formats

Table 7.2 summarizes the main points of the five representations of the conceptual model described (the reference to concept map is according to IOV representation). It can be seen how all these different representations formats are complementary as they have different goals. Please, notice also that each Figure has been manually translated from the source terms in Spanish into English to make the reading of this chapter easier for non-Spanish native speakers. The Spanish terms are gathered in Appendix E.

CONCEPT MAP	
Definition	Graph with nodes representing the concepts and links among them.
Keypoints	Relationships between the concepts and level of knowledge according to the background color schema of the nodes.
Goal	To follow the evolution of the concepts and their relationships during the course.
Figure	Figure 7.4
DIAGRAM	
Definition	Hierarchical diagram with the AC at the top and TCs and BCs below.
Keypoints	The hierarchy of concepts from more to less relevancy and organized by topics from less to more CV as indicated by the background color and the numerical value in a tooltip.
Goal	To have an indication of the level of understanding of the AC and which concepts should be reviewed.
Figure	Figure 7.7
TABLE	
Definition	Set of cells occupied by the BCs organized in rows and columns.
Keypoints	Each row is a BC and the columns show its name and values for the weight and CV.
Goal	To have an accurate justification of the CVs of the BCs according to the scores provided by a free-text ACAA system and their frequency of use.
Figure	Figure 7.8
BAR CHART	
Definition	Bar chart ordered according to the relative percentage of CV of the BCs.
Keypoints	Each row is a BC that is colored according to the percentage of CV that covers regarding the total 100%.
Goal	To compare BCs and ordered them according to their relative percentage.
Figure	Figure 7.9
SUMMARY	
Definition	Three ordered lists indicating the ten best and worst understood BCs and their relevancy.
Keypoints	Each line represents a BC with its name, numerical CV and weight. In the first list, the CV is also indicated by the font color (it is not done in the second and the third list to avoid having a list full of red or green color lines).
Goal	To find out how well the most important BCs are being assimilated.
Figure	Figure 7.10

Table 7.2: Comparison of several representation formats of the conceptual model. The Figures indicated per each of them represents the same conceptual model of a student enrolled in an Operating System course.

Part III

Evaluation and conclusions

In this part, it is presented the evaluation performed to test the Natural Language Processing techniques used for free-text scoring, the effect of the Adaptive Hypermedia techniques and the feasibility of the combination of all these techniques to generate valid conceptual models. Notice that I have considered that a valid conceptual model is the one that better reflects the real knowledge of a student. A special relevance has also been given to the satisfaction degree achieved by students and teachers when using Willow and COMOV. The results and the conclusions drawn from these results end this chapter together with the lines of future work.

This part consists of two chapters (the list of publications that has produced the work described in each chapter is also given):

- Chapter 8 entitled "**Experiments and evaluation**" validates the generated conceptual models from the point of view of teachers and students. Besides, it also gathers the description of the experimental settings, the corpus used and, the results achieved to fulfil the rest of subgoals pursued. The publications related to this Chapter are: Pérez-Marín et al. [2007a,b,c], Alfonseca and Pérez [2004b], Pérez et al. [2005a,b,d,c].

- Chapter 9 entitled "**Conclusions and future work**" highlights the main ideas of this work and prospective lines of future research. It is important to notice that, as explained in the introduction, this is an interdisciplinary work and thus, there have been contributions to several fields (particularly Adaptive Hypermedia, Natural Language Processing and Student Modeling). This Chapter also explores how this work has contributed to each of them, which problems have been solved, and how the proposed procedure can be adapted to be used in other language and/or area-of-knowledge.

Chapter 8

Experiments and evaluation

Once the proposed procedure to automatically generate and display conceptual models from the students' answers has been detailed, and the Will tools have been described, this chapter focuses in proving the feasibility of the procedure to automatically generate students' conceptual models and, the validity of the generated conceptual models by the results achieved in several experiments performed.

In particular, three different experiments in the Operating Systems subject of the Telecommunications Engineering degree at the Universidad Autónoma of Madrid have been carried out during the 2005-2006 and 2006-2007 academic years:

- **First one (December 2005)**: 32 students (75% of the enrolled students) volunteered to use Atenea or Willow during a class with teachers who have previously tested Willed. Students were separated as follows: 16 students used Atenea and 16 students used Willow. One of the goals was to find out the satisfaction degree that students reach when using these free-text scoring systems and, the effect of adaptation by comparing how they use the non-adaptive free-text CAA Atenea to how they use the adaptive free-text CAA Willow system. Furthermore, the main goal was to have the necessary information to generate, for the first time, students' conceptual models and displayed them to teachers using COMOV, which was also tested. That way, the generated students' conceptual models were validated from the point of view of the teachers.
- **Second one (December 2005):** 7 students (out of the group of 32 of the first experiment) volunteered to use Atenea or Willow during a week from any computer connected to Internet. They had the possibility of using the systems without any restriction. The goal was to find out the way in which students use these systems. This is very important as the freely use of Willow (i.e. not during a class) is the key point to the procedure of the automatic generation of the students' conceptual models.
- **Third one (September 2006-January 2007):** 24 students (50% of the enrolled students) volunteered to freely use Atenea or Willow during the whole course. Moreover, in this experiment, both the students and the teacher were given the possibility of keeping track of the evolution of the students' generated conceptual models during the whole course. That way, students could also validate them.

	Author1	**Author2**	**Author3**	**Author4**	**Author5**	**Author6**
Very familiar		x			x	
Familiar						x
Medium-level of familiar			x	x		
Slightly familiar						
Not familiar at all	x					

Table 8.1: Degree of familiarity of the authors with authoring tools.

Finally, in order to complete testing the goals pursued in this work, i.e. to find out the optimum combination of NLP techniques to improve the accuracy of the free-text scoring, which permits the automatic assessment of open-ended questions, another set of experiments and their results are also presented at the end of this Chapter.

8.1 First experiment: using Atenea or Willow during a class and validating the generated models

First of all, a group of six different teachers of the Computer Science department of the Universidad Autónoma de Madrid, whose familiarity with authoring tools is represented in Table 8.1, tested the usability of the authoring tool Willed to enter the information necessary to create the domain model. The procedure followed to perform this test consisted in asking each teacher to complete three tasks with the authoring tool: to insert a new question in one course, to create a new set of questions for an area of knowledge and, to update the information about a question in one course.

Next, they were asked to fill in an anonymous and voluntary Likert-type questionnaire whose results are gathered in Figure 8.1 (see Appendix B to have a look at the questionnaire). It can be seen that all of the interviewed authors, irrespectively of their degree of familiarity with authoring tools, have stated that they would rather use the authoring tool than not use it. Besides, they consider it as very easy (67%) to use, more than 80% think that it is very useful and intuitive, and most of them (67%) claimed that none of the proposed tasks was difficult to complete.

Once Willed had been tested, two teachers of the Computer Science department used it to introduce twenty different questions of different levels of difficulty and topics from real exams of previous courses. The score to pass a question was set to 50% of the maximum score, and the percentage to be promoted or demoted was set to 40% of the total number of questions.

Students' participation in the experiment with Atenea or Willow was voluntary, and the teachers motivated the students by telling them that the questions had been taken from previous exams and, that the practise would positively help them towards the final score in the subject. That way, a total of 32 students (75% of the enrolled students in the subject) took part in the experiment, from which two subgroups were randomly created each one with 16 students: **group A that used Atenea, and group B that used Willow**. Please, notice that as it was the first experiment in which Atenea or Willow were used, the most simple NLP configuration

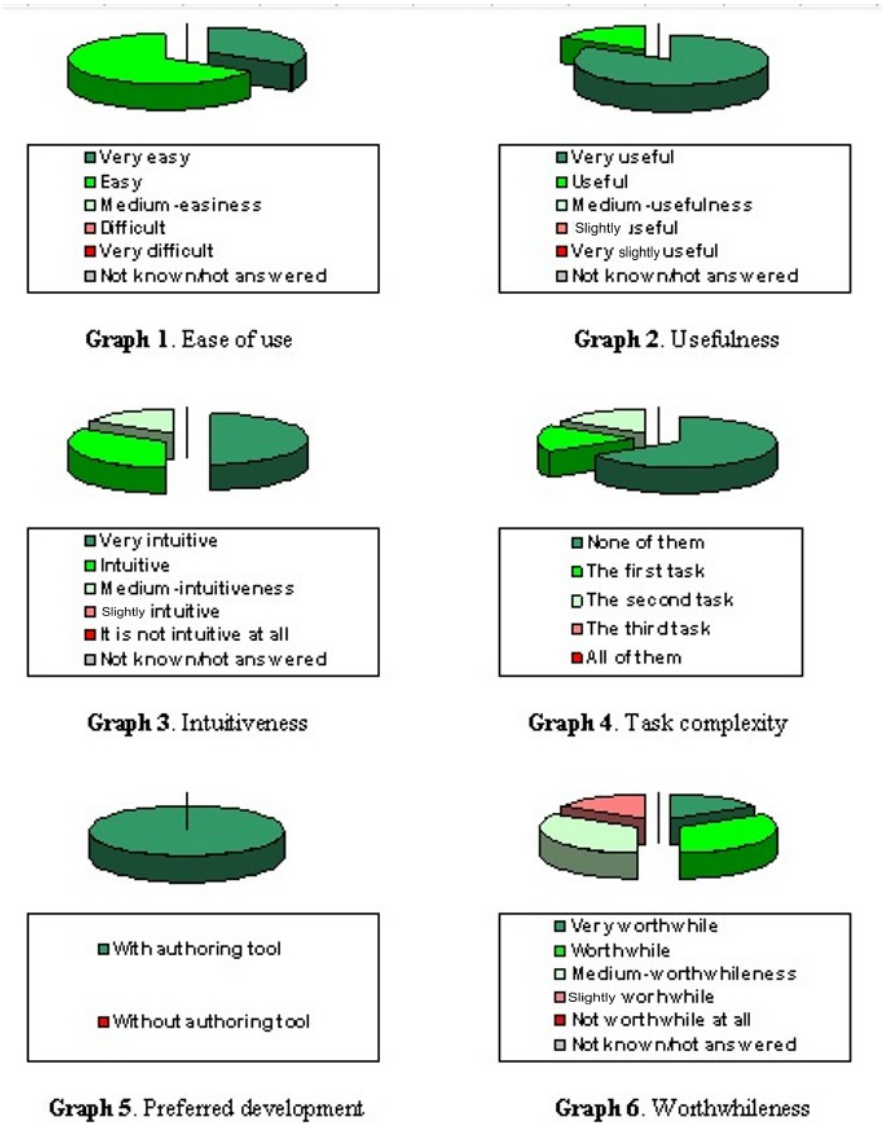


Figure 8.1: Results of the survey carried out about the usability and usefulness of Willed.

was tried and, only the comparison module with ERB was activated.

At the beginning of the experiment, all the students received a brief explanation (5 minutes) about Atenea and Willow, its aim and how to use the system. Next, they were asked to take a 5-minute test with five multiple-choice questions corresponding to the five topics under assessment. In a 0-5 scale, the average score was 2.8 for group A, and 3.2 for group B. Once the test was finished, the students were allowed to start using the indicated version of the system during 20 minutes. After that, they were asked again to complete the same test to check if they had acquired new knowledge during the assessment session. The average score for group A did not change at all, whereas the average score for the group B (who use the adaptive version) increased slightly up to 3.4. Finally, the students were asked to fill in a Likert-type scale items satisfaction questionnaire (see Appendix B to have a look at the students' satisfaction questionnaire). The results are summarized in Table 8.2. It can be seen that:

Question	group A	group B
Familiarity with on-line applications	4.3	3.8
Difficulty of use	4.1	4.1
Intuitiveness of the interface	4.0	3.5
System's answer time	4.1	3.8
Fitness of students' needs	3.4	3.2
Order of the questions	3.2	3.4
Level of difficulty	2.3	2.9
Number of references	3.0	3.0
Number of questions answered	7.0	8.5
Time to study this course	less than 5 h.	less than 5 h.
Recommendation of using Atenea/Willow	yes	yes

Table 8.2: Average results for the first experiment in which a group of students (A) used Atenea and, another group of students (B) used Willow in one of their classes.

- **The usability of Atenea and Willow**: Both groups are quite familiar with on-line applications since the scale goes from 1 (not familiar at all) to 5 (very familiar) and they have high values. Some students also knew about on-line tests, although none of them had previously used either a free-text CAA system or a free-text ACAA system before. Concerning the difficulty of using the system, evaluated from 1 (very difficult) to 5 (very easy) both groups had an average value of 4.1, that is, very easy. When evaluating the interface from 1 (no intuitive at all) to 5 (utterly intuitive), it can be seen that Atenea's interface is slightly considered more intuitive. This result could be due to the fact that it is simpler as it does not show any information about promotions-demotions and there are less buttons as all the options are fixed. Besides, the students who use Willow had not been explained about how to personalize the interface and none of them changed any interface default value, so they could not benefit from using the adaptive interface. Finally, concerning the delay to get the corrected answer, the students agreed that the system is quite fast.
- **The fitness to the students' needs**: In a scale from 1 (it does not fit my needs at all) to 5 (it completely fits me), both groups answered with an intermediate value.
- **The order of the questions**: Group B has returned a slightly higher score concerning the order of the questions, in a scale from 1 (completely dislike the order) to 5 (completely like the order), but not statistically significant. Moreover, when evaluating the level of difficulty of the questions from 1 (very difficult) to 5 (very easy), again group B returned a higher value. Therefore, it is important to highlight how the students that used the adaptive version (Willow) were able to notice that the level of difficulty was not so hard, specially at the beginning, where easier questions were presented.
- **The number of questions answered**: Students in group A have answered an average of 7 questions while the students in group B have answered 8.5. This result was expected as the students of the adaptive version were presented easier initial questions and, thus, they should be able to answer more questions.
- **The number of references**: Nearly all the students stated that between two and four

Feature	T1	T2	T3	T4	T5	T6	Mean
Familiarity with conceptual models	3	2	4	1	4	2	2.7
Intuitiveness of the COMOV's interface	4	4	5	3	3	4	3.8
How informative is the table representation	5	4	3	2	3	5	3.7
How informative is the bar chart representation	2	5	5	4	5	4	4.2
How informative is text summary representation	4	4	5	3	4	5	4.2
How informative is the concept map representation	4	4	5	4	4	5	4.3
Favorite representation	T	B	S/C	B	C	C	C
COMOV usefulness	2	4	5	4	4	5	4.0
Would you use COMOV?	Y	Y	Y	Y	Y	Y	Y
Would you recommend COMOV?	Y	Y	Y	Y	Y	Y	Y

Table 8.3: Results of the satisfaction survey carried out for six teachers of the Universidad Autónoma de Madrid and their average values in the mean column. Notice that the concept map representation is marked as C, the table as T, the bar chart as B and the text summary as S.

is their ideal number of reference answers. In fact, all the students considered as the best feature of the system the possibility to read answers written by the teacher, and to know how they are expected to answer. It is a good result taking into account that it matches the need of having at least three different references to improve the accuracy of the automatic free-text scoring as it will be explained later.

- **The time devoted by the students to use the system**: Most of the students (69% of the students in both groups) said that they cannot review more than 5 hours per week. In fact, due to this lack of time in both groups the trend is just to focus on the topics that have been less understood when presented in the theory lessons.
- **Whether they would recommend to use Atenea or Willow**: 91% of the 32 students would recommend the version used to a friend of Operating System or another subject. In fact, some comments given by the students of group A (non-adaptive version) were *"I think that it is a useful system to learn and review concepts"*, *"Atenea helps you to review, only reading the books is boring...and you cannot auto score yourself!"* and *"I believe it is a good assessing method, it guides you and at the same time it requires that you make an effort to keep answering"*. While some comments given by the students of group B (adaptive version) were *"It allows you to know how clear you have the concepts of the subject and how they are approached by the teachers"*, *"It is a more amusing way to learn"* and *"It can be useful to me or to anybody who wants to learn concepts in an interactive way"*.

Furthermore, from the answers provided by the students to the systems, 31 students' conceptual models were generated according to the proposed procedure[1] and displayed to teachers to find out their opinion about COMOV and to quantitatively and qualitatively validate the models. Thus, six teachers of the Computer Science department at the Universidad Autónoma of Madrid were asked to use COMOV and fill in a satisfaction questionnaire with some Likert-

[1]One student did not take the final exam and thus, his conceptual model was not generated because it could not be compared with a score in the final exam.

type items in a scale from 1 (very negative value) up to 5 (very positive value) and some free-text items. The results of the questionnaire are gathered in Table 8.3 in which each column refers to a teacher and each row refers to an issue addressed in the questionnaire.

As can be seen, most teachers are quite unfamiliar with students' conceptual models. Only two teachers stated that they were familiar with the use of concept maps. In fact, most of them said that they have heard something about concept maps and student models but nothing about automatic generation of students' conceptual models. All the same, they considered COMOV useful to identify how well the students have understood the concepts of the lesson. In fact, the average value is 4.0. Besides, they thought it was very simple to use and the intuitiveness of its interface was given an average value of 3.8.

In general, teachers were positively surprised by the possibility of having an immediate representation of their students' conceptual models. Moreover, all of them would use it in their courses and would recommend its use to other teachers to have more feedback about their students. One of the best regarded options was to have the class conceptual model and to see the general values for all the students. For instance, some comments of the teachers were "*I would use COMOV in my classes because it helps me to know the average knowledge level of my class, both at the beginning and at the end of the course*"; "*It is a fast way to identify which concepts the students understand better and which worst. In general, to see what the students are learning*" or "*COMOV is helpful as feedback for me to see which topics/concepts should be reviewed in the lessons*".

It is important to notice that none of them would be interested in having a qualitative score from any of the representations of the conceptual model with a summative purpose. When the teachers were asked which representation they considered to be the most informative, the average values were very similar, just with a few decimals of difference between them (the table got 3.7 average value, the bar chart 4.2, the summary 4.2 and the concept map 4.3). It might be because they all thought that these representations are quite illustrative of the students' conceptual model (see also that in a scale from 1-5, the average values are always above 3 and quite near 5), and some of them thought that even complementary. The following are the strong points of each representation, as indicated by the teachers (the conceptual diagram is not included as it has been developed after this experiment was performed):

- The concept map details the students' conceptual models and groups them by topics and AC.
- The table gives fast access to conflictive concepts.
- The bar chart is a simple and pleasant way to see the results.
- The textual summary is useful to easily identify the best and worst BCs.

Nevertheless, when the teachers were asked to choose one of the several representations, there were 4 votes for the concept map, 2 for the bar chart, 1 for the table and 1 for the summary (there were 2 teachers who felt unable to choose between concept map and summary so they gave one vote to both of them) so, the concept map can be highlighted as the most representative.

Finally, a quantitative and qualitative comparison of the generated students' conceptual models and the final scores achieved by the students in the final exam of the subject was done.

Info	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Sc.	8	8	9	7	6	6	5	6	7	7	5	6	7	5	8	6
Nq.	5	10	4	6	4	6	5	5	5	4	7	4	5	5	5	8
A	3	1	3	3	1	2	3	2	2	3	2	3	1	1	3	2
B	3	2	6	4	3	2	4	4	3	5	2	6	1	1	8	3
C	8	9	4	5	8	6	7	7	8	6	7	6	10	1	4	7
D	4	1	6	3	2	4	3	4	4	4	4	4	0	0	6	4
E	0	2	1	2	1	0	2	0	0	2	0	1	0	0	2	1
F	.4	.4	.6	.5	.4	.4	.5	.4	.4	.5	.4	.5	.4	0	.6	.4

Table 8.4: Results of the analysis of the first sixteen generated students' conceptual models.

Info	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	ALL
Sc.	6	7	8	5	6	5	8	3	6	7	8	6	7	9	5	7
Nq.	5	6	13	4	6	5	6	7	5	5	6	5	5	6	7	179
A	1	2	2	2	1	2	1	1	2	2	3	1	2	2	2	2
B	4	3	2	2	3	3	5	1	3	3	5	2	5	4	3	4
C	7	8	8	9	7	8	7	10	8	9	5	8	6	7	9	6
D	4	3	4	2	1	4	3	2	2	3	5	2	5	4	3	5
E	1	1	0	1	3	0	2	0	2	0	1	1	0	1	0	0
F	.4	.4	.4	.4	.4	.5	.5	.3	.4	.5	.5	.4	.5	.5	.4	.4

Table 8.5: Results of the remaining 17-31 generated students' conceptual models.

In particular, the students' textual summary and concept map representations were analyzed and the results are summarized in Tables 8.4 and 8.5. Each column refers to a student, and the last column describes the whole class[2]:

- **Sc.**: final theory score given by the teacher (0-10 numerical scale).
- **Nq.**: number of questions answered by the student.
- **A**: number of BCs of the first list in the textual summary, that is, with confidence medium or high.
- **B**: number of BCs in the third list of the textual summary, that is, the ten terms that are best known by the students.
- **C**: number of BCs in the concept map with low confidence.
- **D**: number of BCs in the concept map with medium confidence.
- **E**: number of BCs in the concept map with high confidence.
- **F**: final score of the concept map calculated as the mean of the confidence-values of all the BCs.

It can be seen that, regarding the textual summary, there is a statistically significant positive correlation (30%, p=0.0981) between the score given by the teacher and the number of BCs of the first list of the textual summary, i.e. the most important terms that students should know with a confidence medium or high (row A). Moreover, the correlation is higher (46%, statistically significant, p=0.0101) when it is calculated between the score given by the teacher and the number of BCs in the third list of the textual summary, i.e. the best-known terms

[2]Notice that, per concept, low confidence is fixed below 0.4 CV, medium confidence is fixed between 0.4 and 0.6 CV and high confidence is fixed above 0.6 CV.

by the students (row B). When the concept map was analyzed, the number of BCs with low confidence-value is in row C, with medium confidence-value in row D and with high confidence-value in row E. A possible formula to calculate the final score of the concept map could be to calculate the mean of the confidence-values of all the BCs in the map. In this way, the correlation between the score given by the teacher and this score for the concept map (row F) is **50%** (statistically significant, p = 0.0068).

It is also important to bear in mind that the purpose is not summative assessment. In fact, teachers stated that they would use the generated conceptual models as formative assessment (the goal intended). Besides, Novak et al. [1983] showed that mapping scores were not significantly related to students' Scholastic Aptitude Test (SAT) scores. These findings suggest that a concept map taps into a substantially different dimension of learning than conventional classroom assessment techniques. This said, the fact that the quantitative metric measured correlates positively indicates that the conceptual model is correctly capturing the students' knowledge.

On the other hand, the qualitative approach is more adequate to observe how students with a high score have a more complex conceptual model, with more well-known concepts and links between them. Figures 7.2 and 7.3 contrast the typical concept map generated for a student (or group of students) with a high final qualification and a student (or group of students) with a low score. Teachers can easily discern just by looking at the shape of the concept map how well the student has assimilated the concepts exposed in the lesson. A simple concept map with a few TCs grouping BCs of different TCs and a few type 3 links between BCs indicates a poor understanding, while more TCs and type 3 links indicates a better understanding.

8.2 Second experiment: using Atenea or Willow during a week

As the first experiment was performed during one of their lessons, the students could use Atenea or Willow only for a very short time. Besides, given that it was the first time both systems were tested, students were only allowed to use a limited set of options without having the possibility of changing any of them. Therefore, it was considered relevant to perform a second experiment with the same group of students, but now students were given more freedom and time to use these systems. In particular, they could use Atenea and/or Willow during a week from anywhere, at anytime, and feel free to choose any option. Moreover, they were encouraged to compare Atenea and Willow and fill in a comparison questionnaire by the end of the week. Besides, Willoc was used to change the combination of NLP techniques activated in Atenea and Willow (please, notice that the free-text scoring engine is the same for both systems). In particular, the stemming and ERB techniques were used.

Just the same as in the first experiment, it was not compulsory for the students to participate in this second experiment, and they were given the same motivations to get involved. In total, seven students volunteered to take part in the experiment and six of them filled in the questionnaire. The results are as follows: all the students agree that Willow fits better their needs; they think that the promotion-demotion feature is quite good; and they like the schema

General information from the 7 students that took part in the second experiment:
It has been registered a total of 18 sessions: 16 adaptive and 2 non-adaptive.
About the feedback:
In 5 sessions the feedback selected was just to have the score.
In 3 sessions the feedback selected was the score and the processed answer.
In 10 sessions the feedback selected was the score, the processed answer and the references.
In 17 sessions the students have chosen to see the feedback of questions previously answered but failed.
About the end-of-session condition:
In 18 sessions the end-of-session was simply by closing the application.
In 0 sessions the students choose to show the chronometer.
About the personalization options:
In 6 sessions the background was changed.
In 2 sessions the font family was changed.
In 0 sessions the font size was changed.
In 1 session the statement font color was changed.
In 1 session the answer font color was changed.
In 0 sessions the background answer font color was changed.
In 0 sessions the text area size was changed.
In average each student:
Has logged in 2.57 sessions: 2.29 adaptive sessions and 0.29 non-adaptive.
Has answered 4.71 questions per session: 4.91 in adaptive sessions and 1 in non-adaptive sessions.
Has spent 81.89 seconds to answer a question: 86.56 in the adaptive and 30.5 in the non-adaptive.
Has repeated the same question 1.34 times: 1.37 in the adaptive versions and 1 in the non-adaptive.

Table 8.6: Summary of the logs gathered in the second experiment in which students could use Atenea (non-adaptive sessions) or Willow (adaptive sessions) without any restriction during a week.

of starting with easy questions and next having them increasingly harder. In particular, some comments given by them are: *"The adaptive version fits better my needs as it makes me correctly answer and the order of the questions is more adequate"*, *"The non-adaptive version is more simple but the adaptive version controls my progress"* and *"The level of difficulty of the questions in the non-adaptive version is higher because it does not take into account what you know"*.

Table 8.6 gives a summary of the logs gathered in this experiment. It is important to highlight how students prefer the adaptive version as from the 18 sessions logged, 16 were adaptive sessions. On the other hand, the logs confirm the trend detected in the first experiments of not using the personalization options. In fact, only 33% of the sessions the background or font colors were changed.

Regarding the feedback, in more than 50% of the sessions all the options were chosen: the score, the processed texts and the teachers' references; and in nearly all the sessions they wanted to see the feedback of questions previously answered but not passed (it was the default option). Finally, the logs indicate that none of the students have used the end-of-session by time or by number of questions and that they prefer to simply close the system.

General information from the 24 students that took part in the third experiment:
It has been registered a total of 92 sessions: 91 adaptive and 1 non-adaptive.
About the feedback:
In 0 sessions the feedback selected was just to have the score.
In 6 sessions the feedback selected was the score and the processed answer.
In 86 sessions the feedback selected was the score, the processed answer and the references.
In 92 sessions the students have chosen to see the feedback of questions previously answered but failed.
About the end-of-session condition:
In 6 sessions the end-of-session was simply by closing the application.
In 3 sessions the students choose to show the chronometer.
About the personalization options:
In 18 sessions the background was changed.
In 3 sessions the font family was changed.
In 90 sessions the font size was changed.
In 1 session the statement font color was changed.
In 8 session the answer font color was changed.
In 0 sessions the background answer font color was changed.
In 91 sessions the text area size was changed.
In average each student:
Has logged in 5.41 sessions: 5.35 adaptive sessions and 0.06 non-adaptive.
Has answered 11.40 questions per session: 11.43 in adaptive sessions and 1 in non-adaptive sessions.
Has spent 124.89 seconds to answer a question: 138.10 in the adaptive and 177.00 in the non-adaptive.
Has repeated the same question 3.28 times: 3.31 in the adaptive versions and 1 in the non-adaptive.

Table 8.7: Summary of the logs gathered in the third experiment in which students could use Atenea (non-adaptive sessions) or Willow (adaptive sessions) without any restriction during the whole course.

8.3 Third experiment: using Atenea or Willow during a whole course and validating the generated students' conceptual models

Once the students' conceptual models had been validated from the point of view of the teachers, a third experiment was performed to find out the opinion of the students. Moreover, to give the option to teachers and students of keeping track of the evolution of the students' conceptual models during a whole course (October 2006 - January 2007).

In order to focus the experiment on this goal, the rest of variables in the experiment were kept the same. That is, Atenea and Willow kept the same configuration than in the previous course, the same domain model was used and, again, the students were of the Operating Systems subject of the Telecommunications Engineering degree of the Universidad Autónoma of Madrid. The participation in the experiment was also voluntary (with the same motivations given by the teacher who, by the way, was a different teacher than the previous year) and, students were given complete freedom to use Atenea or Willow. From the total number of enrolled students, 24 (41%) agreed to take part in the experiment.

The results of this experiment have confirmed the conclusions drawn in the previous experiments. Students have stated how much they have enjoyed using the system to reinforce concepts, and Table 8.7 shows a summary of the logs gathered. As can be seen, students

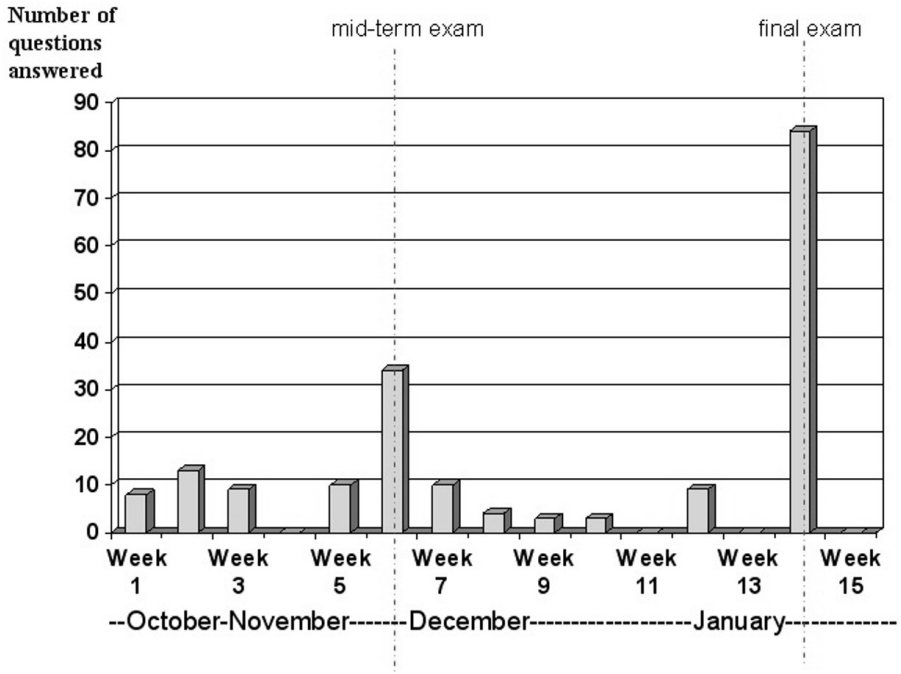


Figure 8.2: Number of questions answered week by week from October 2006 to January 2007 by the students of the third experiment.

massively chose adaptive sessions again, with all feedback. Only six students have used an end-of-session condition different of just closing the application and, in average, they have answered ten questions per session. It is also interesting to observe how these students have exploited more the personalization features as nearly all of them have changed the text area size and the font size. It may be due to the fact that in this experiment, all the information about the systems, even a short tutorial about them was provided to the students.

All the same, it is important to highlight how the number of sessions before important exams dramatically increases as can be seen in Figure 8.2. Week 1 corresponds to the first week of the experiment (October 16th to 22nd). It is considered that the number of students that entered the system the first weeks is just because they were curious about it. Next, the second peak in the graph is around the weeks 5-7, given that the first exam was on November 28th in the seventh week of course. After that, despite students have been advised to complete at least four questions per week to have a complete training before the final exam, there is less use of the system until the end of December - beginning of January when students are again reviewing, in this case, for the final exam at the end of January. Thus, it can be seen how the system is used by the students to get more training before their exams. Besides, concerning the use of the clarification questions, it is important to highlight that from the 174 questions that the students of the third experiment were able to pass, 74.14% were passed thanks to the use of the clarification questions.

Regarding the new possibility of looking at the representation of their conceptual model, 11 students (46%) used it during the course and reported by mail that they appreciated this extra feedback very much. Some comments provided by the students were "*I cannot imagine*

Sets	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean**
No. cands.	38	79	96	11	143	295	117	117	14	14	92.4
Mean cands.	67	51	44	81	48	56	127	166	118	116	87.4
No. refs.	4	3	4	4	7	8	5	3	3	3	4.4
Mean refs.	130	42	30	64	27	55	71	186	108	105	81.8
Type of quest.	Def.	Def.	Def.	Def.	Def.	A/D.	Y/N	A/D	Y/N	Def.	–
Range of scores	[0,0.5]	[0,0.5]	[0,0.5]	[0,1]	[0,1]	[0,0.5]	[0,1.5]	[0,1]	[0,1]	[0,1]	–
Source language	En.	Sp.	Sp.	Sp.	Sp.	Sp.	Sp.	Sp.	Sp.	Sp.	–

Table 8.8: Answer sets used in the evaluation. Rows indicate: set number, number of candidate texts (No. cands.), mean length of the candidate texts in words (Mean cands.), number of reference texts (No. refs.), mean length of the reference texts (Mean refs.), question type (Def., definitions and descriptions; A/D, advantages and disadvantages; Y/N, Yes-No and justification), range of scores as provided by the instructors and source language in which they were written (En., English and Sp., Spanish).

it was like that...it is amazing how I can easily see which concepts are clear and which ones are not" or "*I think it is very interesting to help me know what to review. Thanks!*". Incidentally, Figures 7.5 and 7.6 in Chapter 7, show the evolution of the conceptual model represented as a concept map of the students who participated in the experiment from October to November. It is important to mention here that as students tend to enter Willow more frequently as the exam date gets nearer, the biggest evolutions always take place when there are exams.

On the other hand, when I asked the students why they like to see their conceptual model, some students' responses were: "*I want to see my progress*", "*To see which my weak points are*" or "*I am curious to see it, I think it is very interesting*". Only one student said that he saw a problem with the conceptual model. It was that he felt embarrassed to see his complete lack of knowledge in Operating Systems. All the same, this student also indicated that he wanted to continue having the possibility of looking at the evolution of his conceptual model to see if he was able to improve his results.

Finally, it is also important to mention that the teacher of the third experiment has also confirmed the conclusions previously drawn. In particular, he stated that he considers very useful to have a system that provides him with more feedback about which concepts have already been understood and which ones should still be reviewed. Besides, he appreciated that it was not only at the level of each particular student but the whole class, giving a special relevance to the possibility of keeping track of the progress of his students by following the evolution of the students' conceptual models.

8.4 Other experiments

Another subgoal of this work, necessary to fulfil the main goal of automatically generating the students' conceptual models from their answers, is to find out which NLP tools should be used (combined or independently) to tackle the automatic free-text scoring problem. In particular, to find a balance between the processing time required and the accuracy achieved.

Hence, a corpus of students' answers was built. It consisted of nine sets of questions transcribed from exams of previous courses. In particular, from exams of the Operating Systems and Object-Oriented Programming courses of the Computer Science department of the Universidad Autónoma of Madrid. Given that all answers gathered were written in Spanish, a set of questions with definitions in English obtained from Google Glossary [http26] was also added (set number 1 in Table 8.8).

The ten sets are described in Table 8.8. They sum up a total of 1929 students' answers. All of them were marked by hand by two different human judges, who also wrote several references per each question in order to cover as much as possible paraphrasing in the students answers when trying to compare them to the references. The three main categories in which the questions were classified are:

- **Definitions and descriptions**, they ask for explanation of concepts, e.g. *What is an operative system?.*
- **Advantages and disadvantages**, they ask for distinguishing good and weak points of a certain topic, e.g. *Indicate the advantages and disadvantages of the token ring algorithm.*
- **Yes/No question**, they just look for an affirmative or negative answer but supported with a textual justification, e.g. *Is RPC appropriate for a chat server? (Justify your answer).*

Furthermore, to test the robustness of the approach when ported across languages, it was also tried to evaluate the answers from the exams automatically translated into English using the Machine Translation system called Altavista [http23]. The Google dataset was also translated into Spanish for the sake of completeness. In this way, the ten sets could be evaluated in both languages.

The metric used to evaluate the goodness of the free-text scoring of the answers of this corpus has been the Pearson correlation as described in Section 4.2.2, filling one vector of scores with the humans' scores and the other with the automatic scores. It is because this metric is one of the most commonly used for the systems which provide scores that are not restricted to integer numbers.

Originally, Atenea was only based on the ERB algorithm (a modification of the Bleu algorithm [Papineni et al., 2001]). Section 8.4.1 presents the correlation values achieved by using this statistical technique based on the Bleu algorithm. Moreover, Section 8.4.2 focuses on how Bleu surpasses other related statistical techniques.

All the same, it was clearly confirmed that a statistical technique cannot be the only one applied to free-text scoring system. Thus, more NLP techniques, in particular the ones included in the wraetlic toolkit were tried as shown in Section 8.4.3 and, since all of these techniques mainly focus on the assessment of the style of the answer, LSA was also applied in order to attempt a more semantically approach. The evaluation of the combination of LSA with the ERB module is shown in Section 8.4.4 and, the combination of LSA with the rest of NLP tools in Section 8.4.5. Section 8.4.6 focuses on the optimum tuning of ERB parameters to find its upper bound. Other techniques attempted were Anaphora Resolution as shown in Section 8.4.7 and Term Identification in Section 8.4.8.

Set	No. of reference texts							
	1	2	3	4	5	6	7	8
1	0.3866	0.5738	0.5843	**0.5886**				
2	0.2996	0.3459	**0.3609**					
3	**0.3777**	0.1667	0.1750	0.3693				
4	0.3914	0.3685	0.5731	**0.8220**				
5	0.3430	0.3634	0.3383	0.3909	0.3986	0.4030	**0.4159**	
6	0.0427	0.0245	0.0257	0.0685	**0.0834**	0.0205	0.0014	0.0209
7	0.1256	0.1512	0.1876	0.1982	**0.2102**			
8	0.3615	0.41536	**0.4172**					
9	0.6909	**0.7949**	0.7358					
10	0.7174	**0.8006**	0.7508					
Mean	0.3736	0.4005	**0.4149**	0.4063	0.2307	0.2118	0.2087	0.0209

Table 8.9: Scores of BLEU for a varying number of reference texts (using the source datasets in Spanish except the first one, which is in English).

8.4.1 ERB

The **first experiment** performed was to find out if the BLEU algorithm as originally described by Papineni et al. [2001] was suitable to assess free-text answers. The insight was that it might be applicable, since the core idea in evaluating Machine Translation systems (the original use of BLEU) remains valid: the more similar the candidate text is to the references, the better it is. Thus, BLEU was evaluated, for each of the datasets described in Table 8.8, by comparing the n-grams from the student answers against the references, and obtaining the BLEU score for each candidate. As it has been stated before, the Pearson correlation value between these automatic scores and the humans' scores was taken as the indicator of the goodness of the procedure. Furthermore, the results for different values in the number of references per dataset and, the length of the n-grams used in the comparison between the student's answer and the references, are provided.

Regarding the number of references, it was varied from 1 up to the maximum number available for each question. Notice that this experiment is quite relevant because, as Papineni et al. [2001] warned, BLEU is sensitive to the number and quality of the references. Table 8.9 shows the results for each of the datasets. As can be seen, in general, as expected, the results improve with the number of references. In general, from these results, it can be assumed that a good number of references is at least three so as to have enough lexical variability contemplated without increasing the matching with wrong students' answers. This is the reason why in all the datasets used in the experiments, there is at least three references per question.

The **second experiment** was focused on the fact that BLEU only takes into account the precision and ignores the recall. It is fine for evaluating MT systems but not for assessing students' answers as there is not penalization for students' texts that do not cover some percentage of the information in the references. Therefore, the original BLEU BP factor has been modified in order to consider the recall too. This new factor has been called: the Modified Brevity Penalty (MBP) factor and the algorithm that uses MBP instead of BP, the Evaluating Responses with BLEU (ERB) (see Section 6.5.2 for more details). Table 8.10 shows that the

Sets	BLEU	ERB+MBP(3:1)	ERB+MBP(2:1)	ERB+MBP(1:1)
1	0.5886	0.5525	0.5392	**0.5976**
2	0.3609	0.4249	**0.5329**	0.5262
3	**0.3693**	0.3615	0.3247	0.3546
4	**0.8220**	0.7674	0.7014	0.8064
5	0.4159	0.6135	**0.6815**	0.6420
6	0.0209	0.1223	0.1730	**0.1756**
7	0.2102	0.2750	0.3609	**0.4247**
8	0.4172	0.4106	0.3887	**0.4308**
9	0.7358	0.7012	**0.7817**	0.6484
10	0.7508	0.6357	0.7564	**0.7645**
Mean	0.4692	0.4865	0.5240	**0.5340**

Table 8.10: Comparison of ERB results when using the MBP with trigrams, bigrams and unigrams or only unigrams against the original Papineni et al. BP factor. The source datasets are all in Spanish except the first one, which is in English.

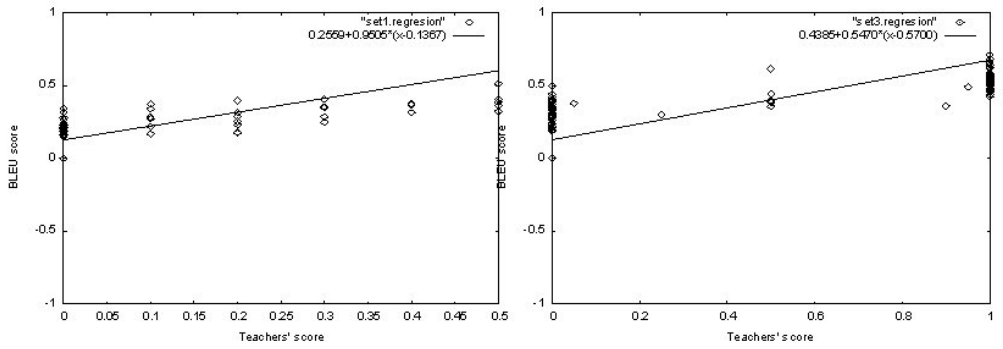


Figure 8.3: Regression lines between the teachers' scores and the automatic marks for sets 2 and 5.

new factor improves the correlation particularly with the datasets with more answers. A t-test proves that the improvement is statistically significative at 0.95 confidence. Another conclusion is that ERB attains the best performance only with unigrams, a result which is very similar to that obtained by Lin and Hovy [2003] for evaluating summaries.

These results suggest that ERB could be used as a lexical comparison module for assessing free-text answers. In order to confirm it, a **third experiment** consisting in two different studies comparing the automatic scores and the teachers' scores for the datasets were done. The first one was based on the use of regression lines while the second one, on histograms.

Figure 8.3 plots the automatic marks against the gold standard for datasets 2 and 5. As expected (because the correlation was positive), the regression line grows with the teacher's score. It can be observed that the teachers' marks are discrete and, in most cases, teachers simply mark the answers as right or wrong. Therefore, most of the points appear either at the leftmost or the rightmost side of the graph. This fact is very evident for set 5, where only a few dots are located at the middle of the image. Still, the automatic scores correlate well with the teacher's scores as can be seen in Table 8.10. For that set, most of the wrong answers received

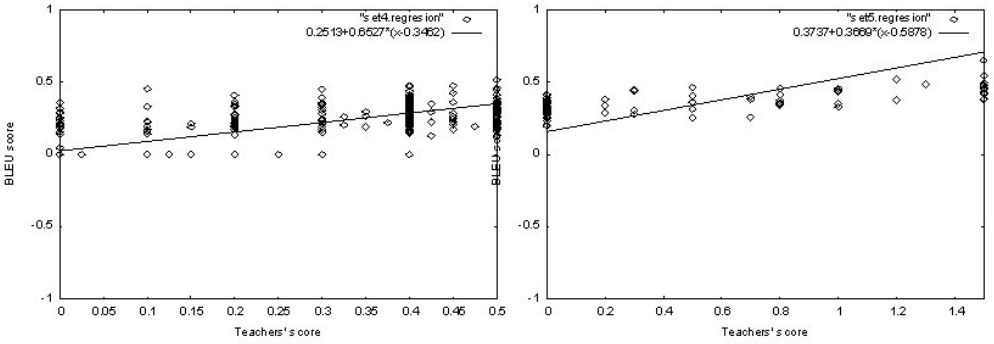


Figure 8.4: Regression lines between the teachers' scores and the automatic marks for sets 6 and 7.

an automatic score below 0.5, and most of the right ones received a score above 0.5, so ERB is distinguishing between right and wrong replies.

Figure 8.4 (left) shows the regression line for set 6. With this set, the correlation is rather poor, as the data points are evenly scattered around the regression line regardless of the teachers' score. This is probably the hardest dataset, as the answers had to include an enumeration of advantages and disadvantages, and ERB is unable to evaluate the discourse structure. For instance, an answer such as "*The use of distributed operating systems is necessary because they do not rely only on one machine but in several. However, it also means that they do not have only one point of failure*" cannot be processed by ERB to extract that "*not to rely on only one machine*" is an advantage and "*to have more than one point of failure*" is a disadvantage. Thus, it is considered that in order to evaluate this question, it will be necessary to perform some other processing such as some kind of rhetorical analysis. In fact, the correlation attained for this set has been the lowest among all sets.

Figure 8.4 (right) displays the regression line for set 7. The results for this set, together with set 8, have been quite surprising. These are yes/no questions in which the students had to justify their answers, so it was expected further pre-processing necessary in order to detect whether the answer was affirmative or negative. However, apparently ERB has been able to correctly deal with these data sets. The answer may lie in the gold standard. If the students' answers are examined, it can be seen that the teachers have not marked the simple yes-no decision, but the student's reasoning supporting it. Many students do get a high mark in the gold-standard even though they have answered wrongly the question, because their reasoning is correct. ERB acts in the same way, disregarding the yes-no, and evaluating the words in the student's discussion.

A **fourth experiment** was about the scale of the output score. ERB always gives a score between 0 and 1, but that is not necessarily the scale used by the teacher who, for instance, may require a question to be marked between 0 and 2 points. Thus, a procedure to scale the ERB's score to the teacher's score must be used. Section 6.5.2 described two possibilities depending on the availability of students' answers of previous courses. Provided that a set of students' answers is available, the regression lines can be used taken the ERB's score as the independent

Set	Regression	Two-points
1	0.81	6.33
2	8.29	15.03
3	6.78	8.50
4	0.59	0.62
5	17.41	22.73
6	25.77	48.08
7	15.59	16.13
8	5.10	29.63
9	0.72	1.31
10	0.19	1.60
Mean	8.13	15

Table 8.11: Mean quadratic error for the several regression lines.

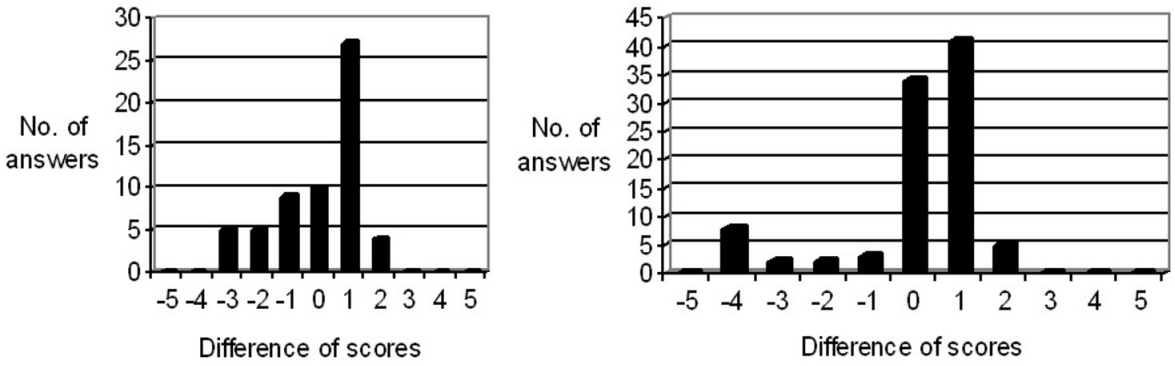


Figure 8.5: Histogram for definition datasets (2 and 3).

variable.

In the case that the set of students' answers is not available, then an estimation of the regression line can be done as the line that crosses the points (0,0) and (1, s_{max}) where s_{max} is the maximum score in the teacher's scale. The cost is that if a student produces an answer that scores best, then the remaining students will see their scores lowered down automatically, as the line will change. Besides, as can be seen in Table 8.11, the mean quadratic errors produced are much higher than using the regression line.

The procedure of the second study consisted in finding out how distant are the teachers' and the automatic scores. Thus, the regression lines were used to scale the ERB output, that is always a score between 0 and 1 to the scale used by the teacher and then, the deviation between ERB's scores and the teachers' was calculated in order to generate the histogram of the deviations. Four histograms are shown here as they represent different types of questions: definitions in Figure 8.5; advantages and disadvantages in Figure 8.6 (left); and, yes / no with justification in Figure 8.6 (right). In all of them, the horizontal axis represents the result of subtracting the teachers' scores from the ERB's ones and then multiplied tenfold. It can be seen that most of the answers either receive the same score or 0.1 points more for definitions, while for other type of questions the results are lower.

These results of the studies based on the regression lines and the histograms indicate that

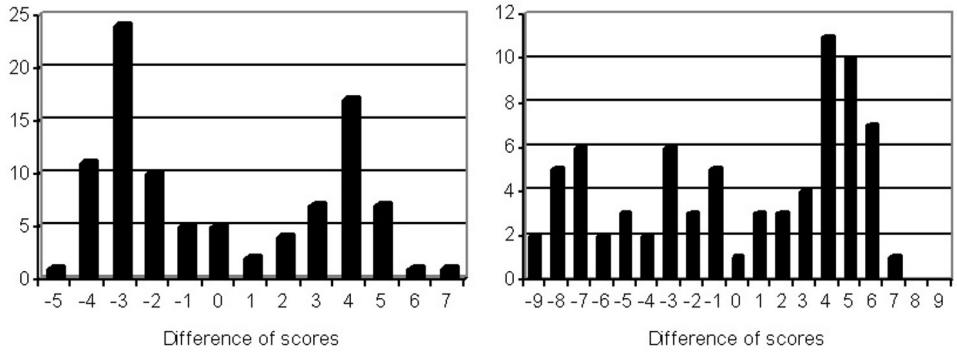


Figure 8.6: Histograms for datasets 5 and 7.

ERB can be used as a lexical comparison module for assessing free-text answers.

8.4.2 Comparison of ERB with baselines

ERB was compared with baseline methods such as Keywords, VSM (that is the one used for lexical comparison in E-rater [Burstein, 2003]) and LSA to have another indicator of its goodness. The results obtained for Spanish are listed in Table 8.12 and, the results obtained for English are listed in Table 8.13.

The **Keywords method** consists in calculating the proportion of words which appear in any of the references by simply counting the frequencies of word occurrences. It is a very simple method that is not longer being used as a reliable technique but still serves as baseline for many applications. For that purpose, it has been implemented here as follows:

1. Initialize a global counter (e.g. $gCounter$) to 0.
2. Calculate the length of the candidate text and store it in $lengthCandidate$.
3. For each word in the candidate text that is found in any reference text, increment $gCounter$.
4. Normalize the result by dividing by the candidate text length, so that longer texts would not be better considered than shorter ones because they have less words:

$$\texttt{Keyword} = gCounter \ / \ lengthCandidate \quad .$$

The **Vector Space Model** (VSM) [Salton et al., 1975] method, although has also received critics [Wong et al., 1987], as for example being too ad hoc a solution, is still being employed in NLP and IR fields. The procedure depicted for evaluating students' answers is the following:

1. Select the training set from the candidate texts.
2. Calculate for each training text its weights, that is, the frequency of each word.
3. For each text vector, transform the weights into tf.idf weights and store the resulting vector into the vector of the training texts vectors, $refWeights$.
4. Calculate the weights vector for the candidate text and store it in a $hCandidate$ vector.

Sets	ERB	Keywords	VSM
1	**0.60**	0.07	0.31
2	**0.53**	0.24	0.10
3	**0.35**	0.20	0.25
4	**0.81**	0.57	-
5	**0.64**	0.57	0.52
6	**0.17**	-0.05	0.05
7	**0.42**	0.33	0.18
8	**0.43**	0.23	0.18
9	**0.65**	0.25	-
10	**0.76**	0.09	-
Mean	**0.53**	0.25	0.23

Table 8.12: Comparison of ERB with two other keyword-based methods. Because of the five-fold evaluation, datasets with very few answers could not be evaluated with VSM. The Spanish source sets are used.

5. `Find out the most similar vector in` $refWeights$ `to the` $hCandidate$ `vector by computing the dot product between each vector of` $refWeights$ `and the` $hCandidate$`, and choosing the one that totals the maximum value.`
6. `VSM = The score of the text that corresponds with the vector chosen of` $refWeights$

A five-fold cross-evaluation has been done, in which 80% of the candidate texts have been taken as training set for calculating tf.idf weights for each term. The rest of the answers are assigned the score of the text in the training set which is most similar to it.

Latent Semantic Analysis (LSA) has been explained in Section 6.5.2. Unlike ERB, it requires an initial (unsupervised) training step. I decided to use two different corpora for the training in order to, not only compare ERB and LSA, but to study how the generality and size of the LSA training corpus affect the results. The corpora used were:

- **CS**: It is a large collection of 142.580 texts from the Ziff-Davis part of the North America Collection corpus. It consists of English extracts and full articles from Computer Science magazines such as *PC Week*, *PC User* or *PC Magazine*, and articles related to Computer Science in more generic journals, such as *The New York Times* or *Business Week*.
- **SA**: It is a collection of 2.902 texts gathered from student answers in an Operating Systems course in the Universidad Autónoma of Madrid. In particular, the translated versions of these Spanish answers to English, using Altavista Babelfish [http23], are used because: CS is in English, an equivalent corpus in Spanish was not available and, as will be proved in Section 8.4.3, to use the translated versions does not decrease much the performance of the system.

It can be seen that, ERB surpasses the results achieved by all baselines. In fact, the improvement is 0.95 significant for all of them. It is also important to highlight that when using the same evaluation framework, the average correlation achieved by using only LSA is lower than by using ERB. Besides, it must be mentioned that using the large corpus (more unrelated to the topic of the students' answers) has given a better result than using the small corpus. This shows that it should be better to use a general corpus about the particular domain in which

Sets	ERB	CS-LSA	SA-LSA
1	0.61	0.49	**0.71**
2	**0.54**	0.20	0.39
3	**0.20**	-0.01	0.17
4	0.29	**0.52**	-0.22
5	0.61	0.50	**0.69**
6	0.19	0.24	**0.27**
7	**0.33**	0.29	0.07
8	**0.39**	**0.39**	0.34
9	0.75	**0.78**	0.66
10	0.78	0.87	**0.91**
Mean	**0.47**	0.43	0.40

Table 8.13: Comparison of ERB with LSA. The English translated sets are used.

the system shall be used, than to try to collect a corpus much more focused on the topic.

8.4.3 NLP+ERB

It is evident that the simple use of ERB (even using its better configuration, that is, unigrams and more than three references per question) is not enough to build a completely new CAA of free text answers system, as it lacks of the necessary level of robustness to deal with problems such as:

- Synonymy, e.g. to detect that "*an operating system is a program*" is equivalent to "*an operating system is an application*".
- Word Sense Identification, e.g. to detect that when "*a process creates a child*" is said, it does not refer to a boy or a girl, but to another process created from the first one.
- Identification of the structure of the discourse, e.g. to detect that in a fragment such as "*distributed systems are useful because the information is shared by several computers, but they have more than one point of failure*", "*to have the information shared*" is an advantage and, "*to have more than one point of failure*" is a disadvantage.

Therefore, it is necessary to complement it with several combinations of NLP techniques as described in Section 6.5.1 and, to find out which is the combinational schema that maximizes the average correlation between the teachers' and the system's scores. In this work, the basic units that have been taken into account for the combinations are: stemming (ST); removal of closed-class words (CC); Term Identification (TI); Word Sense Disambiguation (WSD); replacement of each word by all the related ones in the WordNet synset to which it belongs (ALL); and, use of syntactic dependences (DEPS) as an attempt to discover the logical relationships in the text.

The results are shown in Table 8.14 both for Spanish and English texts. Notice, that the references were translated by hand, while the students' answers were automatically translated using Altavista Babelfish [http23]. In any case, it can be seen that the results do not vary much, which may indicate that the text processing modules do not have a large impact on this task. The best result for Spanish, 54% correlation, is reached by using only ERB and stemming, and for English, 54% correlation too, by using ERB, stemming and removal of closed-class words.

Sets	1	2	3	4	5	6	7	8	9	10	Mean
ERB$_{Sp}$	0.60	0.53	0.35	0.81	0.64	0.18	0.42	0.43	0.65	0.76	**0.53**
ERB$_{En}$	0.61	0.54	0.20	0.29	0.61	0.19	0.33	0.39	0.75	0.78	**0.47**
ERB+											
ST$_{Sp}$	0.62	0.48	0.24	0.68	0.66	0.24	0.40	0.46	0.86	0.73	**0.54**
ST$_{En}$	0.60	0.43	0.17	0.36	0.69	0.24	0.33	0.46	0.81	0.83	**0.49**
CC$_{Sp}$	0.54	0.58	0.32	0.78	0.68	0.24	0.42	0.43	0.69	0.48	**0.52**
CC$_{En}$	0.57	0.55	0.29	0.58	0.61	0.32	0.35	0.45	0.70	0.68	**0.51**
TI$_{Sp}$	0.65	0.47	0.28	0.66	0.63	0.19	0.45	0.41	0.67	0.73	**0.51**
TI$_{En}$	0.61	0.50	0.26	0.30	0.52	0.27	0.30	0.30	0.74	0.83	**0.46**
WSD$_{Sp}$	0.63	0.47	0.22	0.67	0.66	0.24	0.38	0.47	0.89	0.72	**0.54**
WSD$_{En}$	0.62	0.42	0.17	0.40	0.70	0.24	0.32	0.45	0.79	0.84	**0.50**
ALL$_{Sp}$	0.21	0.20	0.11	0.54	0.63	-0.02	0.26	0.29	0.32	0.01	**0.25**
ALL$_{En}$	0.15	0.23	0.08	0.48	0.60	-0.01	0.17	0.29	-0.15	-0.25	**0.15**
DEPS$_{Sp}$	0.32	0.24	0.16	0.65	0.41	0.19	0.13	0.17	0.50	0.55	**0.33**
DEPS$_{En}$	0.41	0.25	0.12	0.60	0.39	0.07	0.16	0.17	0.59	0.51	**0.33**
ST+CC$_{Sp}$	0.58	0.48	0.29	0.81	0.70	0.30	0.40	0.49	0.73	0.48	**0.53**
ST+CC$_{En}$	0.57	0.53	0.28	0.74	0.76	0.35	0.38	0.49	0.65	0.65	**0.54**
CC+TI$_{Sp}$	0.57	0.57	0.34	0.74	0.67	0.24	0.41	0.44	0.65	0.53	**0.53**
CC+TI$_{En}$	0.54	0.57	0.47	0.47	0.60	0.32	0.35	0.35	0.68	0.75	**0.50**
ST+TI$_{Sp}$	0.64	0.47	0.23	0.69	0.66	0.23	0.44	0.46	0.73	0.72	**0.53**
ST+TI$_{En}$	0.62	0.46	0.24	0.42	0.61	0.30	0.32	0.30	0.63	0.81	**0.47**
ST+WSD$_{Sp}$	0.64	0.47	0.23	0.69	0.66	0.23	0.44	0.46	0.73	0.72	**0.53**
ST+WSD$_{En}$	0.63	0.44	0.25	0.38	0.67	0.25	0.30	0.39	0.61	0.82	**0.47**
CC+WSD$_{Sp}$	0.60	0.47	0.28	0.80	0.69	0.30	0.38	0.53	0.77	0.51	**0.53**
CC+WSD$_{En}$	0.59	0.48	0.33	0.71	0.77	0.33	0.36	0.47	0.68	0.68	**0.54**
WSD+TI$_{Sp}$	0.64	0.47	0.23	0.69	0.66	0.23	0.44	0.46	0.73	0.72	**0.53**
WSD+TI$_{En}$	0.61	0.41	0.26	0.38	0.62	0.27	0.29	0.30	0.61	0.82	**0.46**
ALL+CC$_{Sp}$	0.11	0.30	0.14	0.50	0.67	0.05	0.33	0.33	0.32	-0.20	**0.26**
ALL+CC$_{En}$	0.05	0.21	0.20	0.51	0.62	0.02	0.31	0.35	0.05	-0.26	**0.21**
CC+WSD+TI$_{Sp}$	0.59	0.49	0.30	0.77	0.71	0.30	0.41	0.47	0.68	0.54	**0.53**
CC+WSD+TI$_{En}$	0.57	0.44	0.44	0.60	0.68	0.30	0.34	0.35	0.56	0.72	**0.50**
ST+CC+TI$_{Sp}$	0.59	0.49	0.30	0.77	0.71	0.30	0.41	0.47	0.68	0.54	**0.53**
ST+CC+TI$_{En}$	0.57	0.52	0.41	0.65	0.68	0.35	0.37	0.36	0.55	0.72	**0.52**
ST+CC+WSD$_{Sp}$	0.59	0.49	0.30	0.77	0.71	0.30	0.41	0.47	0.68	0.54	**0.53**
ST+CC+WSD$_{En}$	0.59	0.46	0.41	0.60	0.74	0.29	0.34	0.43	0.56	0.73	**0.52**
ST+WSD+TI$_{Sp}$	0.64	0.47	0.23	0.69	0.66	0.23	0.44	0.46	0.73	0.72	**0.53**
ST+WSD+TI$_{En}$	0.61	0.41	0.26	0.38	0.62	0.27	0.29	0.30	0.61	0.82	**0.46**
ST+CC+WSD+TI$_{Sp}$	0.59	0.49	0.30	0.77	0.71	0.30	0.41	0.47	0.68	0.54	**0.53**
ST+CC+WSD+TI$_{En}$	0.57	0.44	0.44	0.60	0.68	0.30	0.34	0.35	0.56	0.72	**0.50**

Table 8.14: Correlations achieved by different combinations of NLP techniques for the original and translated datasets (the comparison module using ERB is always used too). The *Sp* suffix indicates that the set is in Spanish and *En* that it is in English.

It is also important to observe that the difference in the correlations using the original texts and their translations is not statistically significant for any of the configurations tested. My hypothesis was that it is because there is not a great variability in the lexical variability of the original and translated texts. In fact, Table 8.15 shows the number of distinct words found in the students' answers collected for each dataset in Spanish, and the number of words found in the English translations. Notice that the first dataset as was originally in English has not been considered for the Spanish to English translation. It is clear that, for all datasets, there are few less distinct words in the English datasets than in the original Spanish answers. Indeed, the automatic translation has not decreased the variability of the vocabulary much.

Furthermore, when the correlation between the vocabulary reduction and the system's performance is calculated as shown in Table 8.16, there is a positive correlation between the reduction in vocabulary (column DifV) and the decrease of the system's performance (column DifC).

Sets	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean**	**All sets**
Spanish	818	716	332	919	1474	1847	1607	415	342	905	4558
English	674	631	284	781	1174	1541	1337	408	326	770.3	3149
DifV	0.18	0.12	0.14	0.15	0.20	0.17	0.17	0.02	0.05	0.15	0.25

Table 8.15: Number of different words in the Spanish and English datasets. Row DifV shows the percentage of vocabulary reduction due to the translation.

Sets	**DifV**	**DifC**		
		ERB	**CC**	**WSD+CC**
2	0.18	-0.01	0.03	-0.02
3	0.12	0.16	0.03	-0.04
4	0.14	0.49	0.20	0.09
5	0.15	0.07	0.07	-0.07
6	0.20	-0.05	-0.08	-0.02
7	0.17	0.04	0.07	0.03
8	0.17	-0.06	-0.02	0.06
9	0.02	-0.12	-0.01	0.09
10	0.05	-0.01	-0.19	-0.18
Mean	0.15	0.06	0.01	-0.01
Corr.		0.14	0.32	0.14

Table 8.16: Correlation between the percentage of reduction of the variability of the vocabulary and the system's performance for different configurations.

Therefore, it has been observed that the system's performance is very similar when combined with automatic translation, provided that the translation does not reduce greatly the variability in the vocabulary. Therefore, whenever the translated texts do not have a great variability in the vocabulary with respect to the original texts, they can be used instead of the original texts without a significant loss in the system's accuracy of assessment.

8.4.4 ERB+LSA

The complementarity of ERB and LSA (ERB is more focused on a lexical level and LSA on a semantic level) inspired us to combine them. In particular, the four possibilities of combining Bleu or ERB with SA-LSA (the LSA trained with the small corpus) and CS-LSA (the LSA trained with the big corpus) were tried. The results are reported in Tables 8.17 and 8.18. Values highlighted in bold indicate the best combination per dataset.

Table 8.17 shows the performances with α fixed to 0.5, so to assign the same weight to both scores. The correlations achieved clearly show that the combination schema is effective: except for the combination of Bleu and CS-LSA, in all cases the result of the combined system is better than the results as stand-alone applications. Interestingly, when combined to ERB, both SA-LSA and CS-LSA provide the same benefits, even if CS-LSA alone is more accurate than SA-LSA.

To test the upper bound of the combination method, the best parameter settings were also estimated, by optimizing the parameter α on the test set. The mean correlations of the

SET	BLEU	ERB	SA-LSA	CS-LSA	BLEU		ERB	
					SA-LSA	**CS-LSA**	**SA-LSA**	**CS-LSA**
1	0.59	0.61	0.71	0.49	0.69	0.60	**0.73**	0.62
2	0.29	0.54	0.39	0.20	0.40	0.32	**0.54**	0.50
3	0.22	0.20	0.17	-0.01	**0.25**	0.21	0.22	0.17
4	0.73	0.29	-0.22	0.52	0.77	**0.79**	0.12	0.37
5	0.35	0.61	0.69	0.50	0.50	0.40	**0.68**	0.63
6	0.04	0.19	0.27	0.24	0.08	0.05	**0.23**	0.20
7	0.23	0.33	0.07	0.29	0.24	0.25	0.31	**0.35**
8	0.27	0.39	0.34	0.39	0.36	0.30	**0.42**	**0.42**
9	0.09	0.75	0.66	0.78	0.30	0.20	0.77	**0.79**
10	0.26	0.78	0.91	0.87	0.55	0.45	**0.87**	0.85
Mean	0.31	0.47	0.40	0.43	0.41	0.36	**0.49**	**0.49**

Table 8.17: Evaluation of the combined systems fixing $\alpha = 0.5$. Cells report the correlations. Please, notice that these results are for the English translated texts.

SET	BLEU	ERB	SA-LSA	CS-LSA	**BLEU**		**ERB**	
					SA-LSA	**CS-LSA**	**SA-LSA**	**CS-LSA**
1	0.59	0.61	0.71	0.49	**0.73**	0.61	**0.73**	0.61
2	0.59	0.61	0.71	0.49	0.44	0.28	**0.54**	0.38
3	0.22	0.20	0.17	-0.01	**0.25**	0.09	0.22	0.10
4	0.73	0.29	-0.22	0.52	0.60	**0.81**	0.12	0.48
5	0.35	0.61	0.69	0.50	0.59	0.55	**0.68**	0.64
6	0.04	0.19	0.27	0.24	0.13	0.14	**0.23**	**0.23**
7	0.23	0.33	0.07	0.29	0.23	0.33	0.31	**0.38**
8	0.27	0.39	0.34	0.39	0.43	0.45	0.42	**0.46**
9	0.09	0.75	0.66	0.78	0.45	0.62	0.77	**0.81**
10	0.26	0.78	0.91	0.87	0.74	0.84	0.87	**0.90**
Mean	0.31	0.47	0.40	0.43	0.46	0.47	0.49	**0.50**
α					0.30	0.10	0.50	0.20

Table 8.18: Evaluation of the combined systems by optimizing the parameter α. Cells report the mean correlations and the values of α at the bottom. The datasets used are the English translated versions.

combined systems are reported in Table 8.18. In the same table, it is also provided the value of the parameter α exploited to achieve the best results. With this optimization technique, the best combination (ERB and CS-LSA) achieves 50% correlation.

Even if the difference is not very significant, the external large corpus CS used to train LSA has been proved helpful also in combination with ERB. The best accuracy has been obtained by combining ERB and CS-LSA and setting α to 0.2. Notice that a lower weight has been assigned to ERB. In general, it is interesting to highlight that when combining ERB's scores and LSA's scores using the different methods explained before, there is most of the times some slight improvement in the correlation to the humans' scores.

As expected, LSA's accuracy improves when a big corpus is provided for training, even if the SA corpus describes the question domain in much more detail. On the other hand, the benefits of the bigger corpus are sensibly reduced when LSA is combined with ERB. This is quite a relevant point, since it means that, even using a generic corpus (i.e. a corpus that does

Sets	ST+LSA	CC+LSA	ST+CC+LSA	WSD+LSA	WSD+CC+LSA
1	0.63	0.56	0.60	**0.64**	0.61
2	0.45	**0.57**	0.55	0.43	0.51
3	0.21	**0.42**	0.38	0.21	0.39
4	0.39	0.50	**0.56**	0.36	0.52
5	0.65	0.67	0.71	0.66	**0.72**
6	0.24	0.31	**0.34**	0.23	0.33
7	0.30	0.38	**0.42**	0.29	**0.42**
8	0.39	0.43	**0.45**	0.39	**0.45**
9	**0.78**	0.76	0.71	**0.78**	0.72
10	**0.88**	0.78	0.78	0.87	0.77
Mean	0.49	0.54	**0.55**	0.49	0.54

Table 8.19: Results of the combination of ERB+NLP+LSA with $\alpha = 0.5$ and using the CS corpus to train LSA. The English translations of the datasets are used.

Sets	ST+LSA	CC+LSA	ST+CC+LSA	WSD+LSA	WSD+CC+LSA
1	0.64	0.58	0.61	**0.65**	0.62
2	0.34	**0.53**	0.51	0.32	0.46
3	0.14	**0.37**	0.34	0.12	0.33
4	0.49	0.52	**0.57**	0.47	0.54
5	0.65	0.68	0.72	0.66	**0.73**
6	0.27	0.32	**0.34**	0.26	0.33
7	0.33	0.39	**0.42**	0.33	**0.42**
8	0.43	0.44	**0.47**	0.44	0.46
9	**0.81**	0.78	0.74	**0.81**	0.75
10	**0.93**	0.82	0.83	0.92	0.83
Mean	0.50	0.54	**0.56**	0.50	0.55

Table 8.20: ERB+NLP+LSA with $\alpha = 0.174$, 0.346, 0.323, 0.151 and 0.298, respectively. The CS corpus to train LSA is used. The English translated datasets are used.

not include particular references to the questions evaluated), the accuracy for the automatic scoring process can be improved.

8.4.5 ERB+NLP+LSA

Once the combination of ERB and LSA had proved to be useful, the next logical step was to try to incorporate the rest of the NLP techniques. Table 8.19 gathers the results of the combination with α fixed to 0.5 of LSA with the best NLP techniques (i.e. that gave a higher average correlation): stemming (ST), removal of closed-class words (CC), stemming and removal of closed-class words (ST+CC), Word Sense Disambiguation (WSD) and, WSD and removal of closed-class words (WSD+CC). As before, the results are only provided for English as the big corpus (CS, that gave the best results as shown in the previous section) to do the LSA training is only available in English.

Values highlighted in bold indicate the best combination per dataset. It can be seen that nearly all combinations improved from using the techniques independently and that there is no much difference among the different combinations. In all cases, the average has increased a

Sets	No. of refs.	Normal ERB Corr.	Upper Bound
2	22	0.5262	**0.6616**
3	5	0.3546	**0.4720**
5	55	0.6420	**0.8187**
6	20	0.1756	**0.4187**
7	48	0.4247	**0.7025**
8	41	0.4308	**0.7223**
Mean	31.83	0.4257	**0.6326**

Table 8.21: The correlation values obtained while using the original ERB and ERB with the best choice of reference texts that was found by the genetic algorithm. Column 2 shows the number of references that were selected in each case. The datasets used are the original datasets in Spanish.

little, reaching up to 55% in the case of stemming, removal of closed-class words, ERB and CS-LSA. To find the upper values of these combinations, α was optimized and the results achieved are shown in Table 8.20. From the analysis of this table, the conclusions previously drawn are confirmed, as the best combination continues being stemming, removal of closed-class words, ERB and CS-LSA with $\alpha = 0.489$ reaching 56% correlation, which is a state-of-the-art value as shown in Chapter 4.

8.4.6 ERB+Genetic Algorithms

Once ERB had been tested in combination with several techniques and in different configurations, an experiment was done to find out the upper bounds of the algorithm. In order to achieve this goal, it was used the knowledge gathered from the previous experiments. In particular, the fact that ERB is very dependent on the quality of the references written by the teacher. Therefore, the procedure described in Chapter 6 to search among a very large set of possible references was implemented so that the references are not only gathered from the teachers but also from the best students' answers of previous courses. It is unlikely that a teacher would write a larger amount of correct answers for his or her questions. Thus, the correlation obtained can be taken as the upper bound that ERB will hardly exceed in this task.

Table 8.21 shows how the correlation has been improved in all the datasets for which this experiment was done (that is, from our corpus in Table 8.8, the datasets that have more than fifty students' answers). Furthermore, it is important to highlight that while in previous experiments it had been observed that ERB is more appropriate for evaluating definitions than for other kinds of questions, with this procedure, ERB alone is able to reach 70% correlation with a yes/no question and with an enumeration of advantages and disadvantages. Nevertheless, it is important to take also into account that this optimum use of ERB is only applicable when a large corpus of references is available, that is, from the second time a course takes place. In other case, the genetic algorithm would not be correctly trained and the results would be lower.

Sets	FNP	ANP	It
1	0.5799	0.0239	**0.5941**
2	**0.5217**	0.2506	0.5176
5	0.5984	0.5107	**0.6337**
6	**0.1731**	0.0209	0.1529
7	0.2102	0.1878	**0.2222**
Mean	0.4167	0.1968	**0.4241**

Table 8.22: Results achieved using RARE+Willow to reduce the lexical variability. The datasets used are the English translations.

Sets	NGR	ERB+AR	ST+AR	CC+AR	ST+CC+AR	WSD+AR	WSD+CC+AR
1	36	0.5964	0.6141	0.5607	0.5903	**0.6208**	0.6054
2	3	0.5212	0.4688	**0.5824**	0.5501	0.4405	0.4951
5	8	0.6442	0.6355	0.6667	0.7094	0.6537	**0.7199**
6	17	0.2218	0.2370	0.3083	**0.3390**	0.2255	0.3238
7	13	0.2918	0.2853	0.3806	**0.4233**	0.2745	0.4182
Mean	15.4	0.4551	0.4481	0.4997	**0.5224**	0.4430	0.5125

Table 8.23: Results achieved using RARE+Willow to generate new references. Column NGR indicates the number of automatically generated references. The datasets used are the English translations.

8.4.7 RARE+Willow

Other NLP technique tested was Anaphora Resolution (AR) as described in Section 6.5.1 to reduce the paraphrasings and to generate new references. The resources used in the experiments were the AR-engine RARE and the datasets with a large number of references. It is important to notice that since, currently, the only model for RARE is for the English language, the English manually translated versions of the texts were used. Nevertheless, as has been previously proved in Section 8.4.3, it should not affect the results. All the results gathered are shown in Tables 8.22 and 8.23. As before, bold font figures highlight the best results of the possible combinations per dataset.

Table 8.22 shows the correlation values for the first-NP (FNP, in which the NPs are replaced by the first RE which is not the "*it*" pronoun), all-NPs (ANP, in which the NPs in the candidate and reference's texts are replaced by the whole coreferential chain) and, only-it (It, in which only the "*it*" pronouns are replaced by the first RE) methods together with the use of ERB. It can be seen that there is no significant improvement in using RARE and, in some cases, such as in the *all-NPs* method, the correlations decrease for all datasets. Therefore, it is concluded that AR is not useful to improve the results of n-gram co-occurrence similarity metrics.

Table 8.23 gives the result for the second experiment in which the focus is on automatically generating new references by substituting the non-pleonastic "*it*" pronouns with all their REs. It can be seen that although several combinations have been tested, the use of RARE has not been able to improve the 56% average correlation of using ERB, stemming, removal of closed-class words and LSA.

Finally, in order to study the effect of using RARE rather than any other AR-engine, a last

Sets	ST+MGR	CC+MGR	ST+CC+MGR	WSD+MGR	WSD+CC+MGR
1	**0.6332**	0.5643	**0.5959**	**0.6529**	**0.6078**
2	**0.4453**	**0.5677**	**0.4901**	**0.4195**	0.4356
5	0.6563	**0.6277**	0.6906	0.6756	0.7059
6	0.2288	0.2735	0.3192	0.2031	0.2746
7	**0.3449**	0.3126	0.3025	**0.3261**	0.2827
Mean	**0.4617**	0.4692	0.4797	0.4554	0.4613

Table 8.24: Results achieved by Willow using several NLP modules and the method of manually generating new references (MGR). The datasets used are the English translations.

Language	Precision	Recall	F-score
Spanish	0.5974	0.9826	0.7430
English	0.6600	0.8601	0.7469

Table 8.25: Results of using C4.5 (70% training and 30% test) to automatically identify terms.

experiment was performed by annotating the co-referential chains by hand. Table 8.24 shows that the results do not have a dramatic improvement with respect to Table 8.23, and even in some cases the correlation decreased when compared with the results using RARE. The reason could be that RARE is more consistent in its answers (either correct or wrong) than a human annotator.

8.4.8 Automatic Term Identification

The approach proposed treats the problem of extracting terms as a classification task described in Section 5.4.1. In the experiments, the decision tree for the C4.5 algorithm has been trained using the ten original datasets and their translations. This contains 4617 words in Spanish, and 4636 in the English version. This corpus of students' answers is taken as the domain-specific corpus, whereas a collection of news on Computer Science with 50.823 words for Spanish and 157.340 words for English is taken as the generic corpus.

Three different human annotators reviewed by hand the references included in the specific corpora in order to build a gold standard. The criteria agreed to determine that a certain n-gram was a term, was that it was specific to the domain and that it was a noun or a noun phrase. Afterwards, the terms that appeared in the three lists were automatically incorporated in the gold standard, and the three annotators discussed about the discrepancies until an agreement was reached in each case. In this way, a list with 76 terms for Spanish and 89 for English was produced.

For training, the samples were chosen so that the distribution of classes were balanced (50% terms and 50% non-terms). Regarding the features considered as attributes, they were the relative frequency of appearance of the term in a corpus of students' answers with respect to its frequency in the generic corpus and the sequence of part-of-speech tags of the words (e.g. noun, verb, adjective, etc.).

The metric chosen to measure the goodness of the procedure was the F-score (described in

Section 4.2). Taking into account that, in this case, for the precision: the number of correct terms found divided by the number of terms found is considered and, for the recall: the number of correct terms found by the total number of terms. In this way, the results achieved after performing a 10-fold stratified cross-validation are shown in Table 8.25. It can be observed that, even using small corpora (albeit very focused on the topic), it has been possible to reach results that are higher than the results attained by other related systems such as Pantel and Dekang [2001] with 0.6781 F-score for English. Besides, recall is well above precision, which is appropriate given that the list of extracted terms is later manually reviewed by the teacher. Therefore, it is important that most of the relevant terms are identified, as the noise can be removed by the teacher during the manual review phase.

Chapter 9

Conclusions and future work

Teachers want students to acquire certain concepts and their relationships as they are accepted in the context of a course. However, what teachers try to transmit to students and what students are actually able to understand is sometimes quite different [Sigel, 1999]. One reason for this can be found, according to the constructivist view of learning, in that each student is building his or her specific cognitive structure depending on his or her previous knowledge and particular experiences. That is, each student has his or her unique and specific conceptual model, which can be defined as the network of concepts and relationships among these concepts.

Therefore and, according to the Meaningful Learning Theory of Ausubel [Ausubel, 1968], each student will only be able to understand new concepts provided that they have some kind of relationship with previous concepts in his or her conceptual model. However, traditional forms of assessment based on objective testing are unable to detect which each student's conceptual model is and thus, they failed to detect the underlying students' misconceptions [Sigel, 1999].

In the last decades, more sophisticated forms of formative assessment have been devised such as assessing free-text answers [Valenti et al., 2003] or, engaging the students in reflective activities with diagnostic purposes [Dimitrova, 2003]. All the same, none of the existing free-text scoring systems takes into account any kind of student's model and, none of the existing diagnostic tool systems is able to deal with free-text answers to generate a complete student's conceptual model.

In this work, it has been proposed a procedure to automatically generate each student and the whole class conceptual models from the answers provided by the students to a free-text Adaptive Computer Assisted Assessment (ACAA) system. Free-text ACAA systems are the evolution of current free-text CAA scoring systems so that they do not only evaluate the free-text answers automatically, but also tailored to each student's model. This procedure has been implemented in the Will tools that consist of: Willow, a free-text ACAA system; Willed, an authoring tool; Willoc, a configuration tool; and, COMOV, a conceptual model viewer.

Several demonstrations of the Will tools have taken place in international conferences such as the "*22nd International Conference of the Spanish Society for the Natural Language Processing*" and specialized workshops such as the "*International Workshop on Applications of Semantic Web Technologies for E-Learning*". Besides, the Will tools have been tested in a

set of experiments carried out in the Operating Systems subject of the Telecommunications Engineering degree of the Universidad Autónoma de Madrid in the 2005-2006 and 2006-2007 academic years.

That way, the feasibility of the procedure has been proved and, a bridge between what teachers try to teach and what students actually understand as reflected in their conceptual models has been built. In particular, Sections 9.1 and 9.2 focus on the fulfilled goals for teachers and students, and Section 9.3 focuses on other subgoals fulfilled. Next, Section 9.4 highlights the new interesting features introduced by the Will tools with respect to the existing state-of-the-art systems and Section 9.5 explains how the procedure described in this work can be applied to other language and/or area-of-knowledge. Finally, Section 9.6 ends this work by describing the main promising prospective lines of future research opened.

9.1 Fulfilled goals for teachers

Two main goals were highlighted in Chapter 1 for teachers, namely to give them more feedback to know how well the concepts have been understood and to find out the main students' misconceptions, and to be able to keep track of each student and the whole class learning progress.

Regarding the first goal, teachers have been provided feedback to know how well the concepts taught in the lessons have been understood by looking at the representation of the generated students' conceptual models (individual or of a group) as they choose (concept map, conceptual diagram, table, bar chart or textual summary) in conceptual model viewers such as COMOV. Moreover, teachers can identify several types of misunderstandings that have been classified in the taxonomy of detectable errors detailed in Section 5.4.6.

Regarding the second goal, teachers can keep track of the students' learning progress by looking at the representations of the conceptual model several times during the course, i.e. from the concept map, they can easily see the evolution by observing how the new concepts modify the previous ones and the new links that are being created.

Finally, from the experiments carried out, it can also be concluded that:

- Teachers have found COMOV quite useful and usable. All of them stated that they would use it in their lessons to see how well students are understanding the concepts exposed in the lectures. They would also recommend its use to other colleagues in other subjects.
- All of the implemented representations to display the conceptual model (concept map, conceptual diagram, bar chart, table and textual summary) were given high scores (between 3 and 5 in a scale from 0 to 5) by the teachers surveyed. It might be because they are partially complementary, highlighting different aspects of the conceptual model. All the same, concept maps have been chosen by the majority as the most informative representation format of the conceptual model. This supports Novak's claim that concept maps are one of the most powerful tools to visualize conceptual knowledge [Novak and Gowin, 1984].
- Teachers have qualitatively and quantitatively validated the generated conceptual models, as they have had the opportunity of checking how the conceptual model reflects how well

students are understanding the concepts exposed in the lessons. In fact, they have observed how students who achieve high scores in the final exam have more complex concept maps than students who achieve lower scores whose concept maps are simpler. Furthermore, a 50% (p=0.0068) statistically significant correlation has been found between the scores achieved by the students in the final exam and the average value of the CVs of their generated students' conceptual models.

- Teachers have also highlighted the importance of receiving feedback and the possibility of following the conceptual evolution of their students (i.e. the possibility of keeping track of how students are assimilating the concepts).

9.2 Fulfilled goals for students

Three main goals were highlighted in Chapter 1 for students, namely to provide them with a procedure that assesses their free-text answers in an adaptive and automatic way, to allow them to identify their main misconceptions, and to guide the students to the correct answer instead of just giving them the correct answer.

Regarding the first goal, students have been given access to a new type of formative assessment tool: the free-text ACAA system Willow. It gives them as feedback not only a numerical score, but their processed answer with a color code to indicate the strong and weak points of their answer (they are reminded this information whenever a question failed is presented to them again) and the correct answers. In fact, due to the great amount of possible feedback, students can choose whether to have access to all of it or just some elements such as the numerical score but not the processed answers.

Regarding the second goal, students have been allowed to identify their main misconceptions and erroneous links (just the same as the teachers) by having a look at several forms of representation of their conceptual model and the whole class conceptual model with COMOV.

Regarding the third goal, reflective thinking has been fostered with the aid of a set of clarification questions in Willow. That way, whenever a student fails a question, s/he is not directly presented the correct answers, but a first clarification question is asked for a more general explanation of the student. If this question is again failed, a second clarification question is asked about a concept in the conceptual model that has a low confidence value. In the case that this question is also failed, a third and more specific clarification question in the form of a Yes/No question is asked. Only when this third question is also failed, the student is presented the correct answers and the question is marked as failed and to be asked again. Notice that 74.14% of the questions that the students of the third experiment were able to pass was thanks to the use of the clarification questions.

Finally, from the three experiments carried out with Willow, it can also be concluded that:

- Students find useful and easy to use Willow. They highlight that Willow helps them to review concepts.
- Students have appreciated very much to have the teachers' references available in order to know how they should answer the questions.

- Although none of the teachers told them, in any moment, that using Willow were funny, the students highlighted that being trained in this way was interactive and amusing. Moreover, they stated that they have felt more engaged to keep answering questions than if they had to read from a static book and do the exercises by hand.
- Students prefer the adaptive version of the system (Willow) to the non-adaptive version of the system (Atenea). It is because despite in the first experiment with the students the difference between both systems was not statistically significant (notice that they could only use the systems during twenty minutes and with all the options fixed), in the second and the third experiments in which students could use both systems without any restriction from any computer connected to the Internet, students massively chose to use Willow. In fact, most of the students reported that the adaptive version controls better their learning process with the promotions and drops of difficulty level. They also declared that the adaptive version fits better their needs, the order of the questions is more adequate, and that in general, they feel more satisfied with it. Besides, the students who used the adaptive version were able to answer more questions and they got a slightly better score in the post-test.
- About the personalization options, although in the first and second experiments, no student changed any of the default values. In the third experiment, when students had more information available about Willow's features and the whole course (from October 2006 to January 2007) to use it, it was observed how they effectively used them and changed some default values.
- The generated conceptual models have been validated by the students too. They have been able to follow their conceptual evolution during a whole semester and, they have stated how much they have appreciated the extra feedback given.

9.3 Fulfilled subgoals

Three main subgoals were highlighted in Chapter 1, namely to overcome the limitations imposed by traditional objective testing sections in Adaptive Hypermedia systems, to make the assessment of free-text answers adaptive and, to find out the optimum combination of Natural Language Processing techniques to improve the accuracy of the scoring of the free-text answers.

Regarding the first subgoal, this work has focused on the assessment of free-text answers that can be done with stand-alone systems or, integrated with Adaptive Educational Hypermedia systems as explained in Section 6.1.

Regarding the second subgoal, a natural evolution of free-text CAA systems (free-text ACAA systems) has been proposed. The core idea of free-text ACAA systems is similar to the core idea of free-text CAA systems, that is, the more similar the student's answer is to the references or correct answers provided by the teachers, the better it is, and thus, the higher the score it will be given. The improvement of free-text ACAA systems is that:

- They keep a student model. The student model contains a static and a dynamic component. In the static component, the values of the features of the course are stored as

determined by the teacher in the authoring tool. That way, the course is presented to the students adapted to these values. While in the dynamic component, the values that change as the students answer the questions proposed by the free-text ACAA system are stored.

- The dynamic techniques comprise a new promotion-demotion difficulty-level procedure that dynamically changes the order of the questions to ask according to each student's model and, a set of clarification questions to guide the students towards the correct answer according to each student's conceptual model.

Regarding the third subgoal, from the NLP techniques tested, the optimum combination found for Spanish is ERB and stemming, reaching 54% Pearson correlation between the teachers' and Willow's scores for the same sets of questions; and, for English is ERB, stemming, removal of closed-class words and LSA reaching 56% Pearson correlation (both of them state-of-the-art results). Finally, from the experiments performed, it can also be concluded that:

- The only use of ERB is not enough to build a complete stand-alone free-text CAA system but it can be successfully applied to free-text scoring by being included in the system's architecture as a lexical comparison module. In fact, ERB has outperformed other statistical techniques such as keywords, VSM or LSA that are being used in other free-text CAA systems such as E-rater [Burstein et al., 1998] or IEA [Foltz et al., 1999].
- The main weakness of ERB, that is its strong dependency on the quality of the references provided, can be alleviated by using AR techniques to generate new references from the existing ones. Furthermore, by using genetic algorithms to choose the best references from the resulting set union of the best students' answers of previous courses and the correct answers provided by the teachers. In fact, this procedure improved the average correlation up to 63% in the six Spanish datasets in which it was tested and, it was able to attain 70% correlation in advantages/disadvantages and yes/no with justification questions that are the most difficult to address with Willow.
- Results achieved using the English translated texts are valid because the translation has not reduced greatly the variability in the vocabulary, as proved in Chapter 8.
- AR techniques are not useful to improve the results of n-gram co-occurrence similarity metrics such as ERB for free-text scoring.
- The automatic identification of terms as a classification task based on the Quinlan's C4.5 algorithm has proved to be very useful, reaching 0.74 F-score both for the Spanish and English corpora. Besides, recall is well above precision, which is appropriate, given that the list of extracted terms can be manually reviewed by the teacher.
- It is possible to extract high-quality term candidates even from a very small domain-specific corpora (only four thousand words). It may be due to the fact that the references written by the teachers are very high-quality and focused texts, so just a small amount of them provides a good amount of data for the identification.
- It is better to use a larger corpus to train LSA instead of trying to collect a corpus much more focused on the topic in which the system shall be used. In fact, the best results for LSA were achieved using a big Computer Science (CS) corpus.

Id	Feature
1	Support meaningful learning
2	Foster reflective thinking
3	Show the evolution of the students' conceptual models
4	Offer an authoring tool that covers all aspects of the course
5	Mark organization of ideas
6	Give instant feedback to instructors
7	Supply emotional interaction
8	Use an individual and group inspectable students' conceptual model
9	Have multilingual support
10	Generate students' individual or group conceptual models
11	Provide automatic and adaptive assessment of free-text answers
12	Use references in plain text for free-text scoring
13	Avoid the necessity of training for free-text scoring
14	Allow the personalization of the interface
15	Display the students' conceptual model in several knowledge representation formats

Table 9.1: List of features.

- In the case of the English language, in which it was possible to test the combination of ERB and LSA, it was found that this combination achieves better results than using ERB and LSA independently. In any case, training is not necessary, since when LSA is not available, and no training is performed, the results only decrease down to 54% correlation between the teacher's and the system's scores.

9.4 Comparison to related systems

Id	Systems
1	ALE, COMPASS, ConceptLab, CREEK-Tutor, E-TESTER, KBS Hyperbook, LEO, StyLE-OLM, **Willow**
2	ALE, ConceptLab, CREEK-Tutor, DynMap+, KBS Hyperbook, StyLE-OLM, **Willow**
3	ALE, ConceptLab+VisMod, DynMap+, LEO, STyLE-OLM+GISMO, **Willow+COMOV**
4	ALE, COMPASS, DynMap+, E-TESTER, KBS Hyperbook, LEO, TADV, **Willow**
5	AEA, COMPASS, ConceptLab, C-rater, Jess, MarkIT, MRW, RMT, **Willow**
6	ALE, VisMod, DynMap+, E-TESTER, GISMO, TADV, **Willow**
7	KBS Hyperbook, RMT, STyLE-OLM, VisMod, **Willow**
8	ConceptLab, DynMap+, **Willow**
9	IntelliMetric, **Willow**
10	**Willow+COMOV**
11	**Willow**
12	**Willow**
13	**Willow**
14	**Willow**
15	**COMOV**

Table 9.2: Systems from the ones reviewed in Chapters 3 and 4 that fulfil the features identified in Table 9.1.

Fist of all, please notice that all references of the systems cited in this Section are provided in the state-of-the-art review in Chapters 3 and 4 and thus, they will not be repeated here.

Furthermore, notice that since a list of relevant features that all these systems should have is not available, a new list has been made with the features mentioned by one or more state-of-the-art systems. Table 9.1 shows this list, and Table 9.2 the systems in which they are present.

As can be seen, the Will tools offer some desired features that are not present in other systems, such as: generating students' individual or group conceptual models; providing automatic and adaptive assessment of free-text answers (without training and with references in natural language); allowing the personalization of the interface; and, displaying the students' conceptual model in several knowledge representation formats. The most related systems to the Will tools are STyLE-OLM+GISMO [Mazza and Milani, 2004] and E-TESTER [Guetl et al., 2005]. All of them pursue a similar goal: to provide more feedback to instructors so that they can identify the main conceptual weaknesses of their students and, help students to organize their concepts.

However, Willow differs from STyLE-OLM in a very important point: Willow generates automatically the conceptual model from free-text students' answers, whereas in STyLE-OLM the model is negotiated between the system and the student. STyLE-OLM is based on engaging the student in an interactive dialog based on conceptual graphs (without use of any NLP technique). The student's model is an overlay of the domain model previously introduced as an ontology in the system and stored as XML. The resulting negotiated student model is shown with CourseVis as a cognitive matrix and with GISMO as a set of dependencies among concepts or histograms. No concept map, conceptual diagram, table, bar chart, or textual summary is generated.

The main improvement of the Will tools with respect to E-TESTER is that in the Will tools not only the concepts but their relationships are taken into account (i.e. a complete conceptual model is generated), whereas E-TESTER just shows a histogram of frequencies of concepts as used by the student and by the teacher to let the user compare them. Besides, the feature of E-TESTER of generating "*What is XXX?*" questions from the frequencies counted, is also included in Willow. In particular, it is the second type of clarification questions. In fact, Willow does not only ask for definitions, but also for general explanations and yes/no questions with the purpose of guiding the student to the correct answer.

The rest of reviewed systems are only partially related to the Will tools. They all have in common that they use some kind of conceptual model, but their goals are different. In particular, ALE is more focused on allowing students to navigate through an interactive semantic space based on concept maps, which are not generated by the system, but introduced in the system with the authoring tool WINDS. COMPASS, ConceptLab+VisMod, KBS Hyperbook and LEO are more oriented to teach new concepts to the students. Regarding TADV, it is focused on producing advices to instructors (not to students).

Finally, it is important to highlight that all the systems reviewed in Chapter 4 are only devoted to free-text CAA, without taking into account any student's model or using any static or dynamic AH technique to modify the formulation of the questions. This is a problem for on-line formative assessment since the students' personal information (level of knowledge, preferences, etc.) is not taken into account and thus, most students do not feel engage to keep answering.

NLP technique	Spanish	English	Other
ERB*	Yes	Yes	Yes
Stemming*	Yes	Yes	No
Removal of closed-class words*	Yes	Yes	No
Term Identification*	Yes	Yes	No
Latent Semantic Analysis	No	Yes	No
Genetic Algorithms	Yes	Yes	Yes
Anaphora Resolution	No	Yes	No

Table 9.3: Natural Language Processing techniques that can be used by the procedure (the techniques that are essential for the procedure are marked with an asterisk) and the languages in which they are currently available in Willow.

9.5 Extending the procedure to other language and/or domain

The examples given in this work have been taken from the Operating Systems area-of-knowledge, given that I work as a teacher in this area. However, the procedure described can be applied to other areas-of-knowledge following the steps explained in this Section, provided that it is not necessary to assess creative thinking or mathematical calculations, which are completely out of scope of this work. Nevertheless, it can also be applied to other languages different than Spanish or English, just by taking the requirements explained in Table 9.3 into account. The steps are as follows:

1. Provided that the language in which the procedure wants to be applied is English or Spanish, the teachers and students can use the systems implemented without further modification. It is because all systems' interfaces have been designed in both languages. For other languages, the systems' interfaces should be translated into the target language.
2. Next, teachers should ask for a login and a password to access Willed. This can be done by sending a mail to Willed's administrator, as appears in the login page.
3. Once teachers have a login and a password, they have to create a new area-of-knowledge with Willed and afterwards, fill in the forms about its name, description, features and topics that the area-of-knowledge comprises.
4. The name of the course provided by the teacher is stored in Willow's database as the area-of-knowledge (AC).
5. The name of the topics of the course provided by the teacher are stored in Willow's database as the topic-concepts (TCs), and type 1 links between the AC and each TC are created.
6. Next, teachers should introduce the questions using Willed. In particular, it is necessary, per each question: its statement and references in the different versions for each value of the features considered, maximum score, topic and level of difficulty.
7. Regarding the references, it is important to highlight that due to the results achieved in the experiments performed, it is advisable that, there are at least three references per question, written by different teachers, in an attempt to capture as much lexical paraphrasing as possible.

8. English teachers can also ask Willed's administrator to generate new references from the ones typed using Anaphora Resolution. In the case of the rest of languages, this feature is not available yet, because RARE needs to be provided a model of the language that, currently, is only developed for English.
9. English and Spanish teachers do not need to be provided any other Natural Language Processing techniques or resources, and can go directly to the next step. In the case of a different language, it would be necessary to have a stemmer, a Part-of-Speech (POS) tagger, and a specific and generic corpora for the Term Identification Module. It is necessary to classify the candidate n-grams in the references as terms. The specific corpus is given by the references provided to Willed and the generic corpus can be automatically retrieved from the web.
10. Willow's administrator should apply the Term Identification module to the references provided by the teachers to generate a list of terms. This list can be reviewed by the teachers, and the resulting list of terms is stored in Willow's database as the basic-concepts (BCs) of the conceptual model together with their frequency in the teachers' references. Type 2 links between each BC and the TC to which it belongs to are created.
11. Teachers ask their students to register in Willow's system by sending a mail to Willow's administrator.
12. Students answer the questions that are stored in the database using Willow, which is keeping track of each student's use of the BCs found in his or her answers in order to calculate each BC confidence-value according to the RateConfidence and ScoreConfidence metrics explained in Chapter 5. Furthermore, Willow is looking for patterns BC+verb+BC to fill in the type 3 links between BCs for each student's conceptual model.
13. During the course, teachers can see each student's conceptual model or the group conceptual model using COMOV. Students can also log into COMOV whenever they want to see their own student's conceptual model as well as their group conceptual model.
14. For the next course, teachers who are willing to use Willow again can ask Willow's administrator to tune some internal parameters to improve the accuracy of free-text scoring. In particular, from the information gathered this year:
    - Genetic algorithms can be runned on the references and students' answers to choose the best references among this set of texts.
    - The regression lines can be calibrated for scaling the internal Willow's score in the 0-1 scale to the scale provided by the teacher to each question.
    - Teachers can be asked to manually score a set of answers to calculate the Pearson correlation between the automatic and the teachers' scores for these answers. This is because the optimum combination of techniques for a different area-of-knowledge and/or language may be different.

9.6 Future work

I plan to continue improving the procedure to automatically generate the students' conceptual models by:

- Applying the procedure to other area-of-knowledge and/or language different than Operating Systems and Spanish.
- Exploring more sophisticated possibilities of generating the group conceptual model.
- Producing templates of good and bad concept maps, so that teachers can easily recognize them at a glance.
- Including the possibility of providing access to the related content of the course by clicking on the nodes (concepts) or the links (relationships) unknown in the class conceptual model, so that Willow can also teach content that covers the deficiencies found.
- Allowing teachers and students to modify the generated conceptual model.
- Improving the set of clarification questions to foster students' reflection. For instance, by applying Natural Language Generation techniques to create new open-ended questions according to the Willow's evaluation of the conceptual model. In this way, Willow will be able not only to assess free-text answers and build conceptual models, but also to generate new specific questions for each student according to his or her particular knowledge level and features in the static profile. I expect that this line of work culminates with the development of an automatic socratic tutor able to foster meaningful learning by engaging students in a set of open-ended questions based on the conceptual model.

As can be seen, I intend to continue exploiting the possibilities of the combination of AH and NLP techniques and keep improving the Will tools with the comments from teachers and students. In particular by:

- Trying more complex combinational schemas between the ERB and LSA scores and testing them both for Spanish and English.
- Improving the syntactical parser for Spanish and English.
- Implementing the Spanish module for the AR-engine RARE.
- Integrating new techniques such as Information Extraction (IE) and Rhetorical Analysis into the Willow's pipeline NLP processing module. IE techniques are expected to be useful because they have been successfully used in other free-text CAA systems. Rhetorical Analysis could help to identify both in the student's and in the teacher's answer, the fragments of the text in which advantages or disadvantages are given, a concluding remark is provided, etc. Besides, IE techniques can help in improving the extraction of type 3 relationships between basic concepts.
- Including more dynamic adaptation in Willow, such as updating the level of difficulty of each question according to the answers given by most of the students or allowing students to move freely between the questions, with a color code that warns them about whether each question belongs to their knowledge level or not.

Finally, to generalize the procedure to automatically generate conceptual models not only for students but, in general, to any user and in particular, to elders or people with some kind

of mental disability. As it has been shown, Willow has already several personalization features that allow users to change some parts of Willow's interface to make work with it simpler. In particular, the font type can be augmented for elders and, some items could be removed for people with some kind of mental disability.

In general, for users who cannot use Willow because it still remains too complex (e.g. they do not know how to use a web browser), it would be possible to let them write freely. The text gathered will be processed by Willow afterwards, resulting in a process completely transparent to the users, who only see the final result: the conceptual model generated and represented adapted to them. In order to achieve this goal, the following several subtasks have to be done:

1. The text is processed with stemming, removal of closed-class words, Term Identification, Name Entity Recognition and Information Extraction to detect and relate entities in the texts such as people, places or dates.
2. The output of the first subtask is a list with the main terms (concepts), entities and their relationships used in the text provided. From this information, the conceptual model can be generated and represented in a graphical way.
3. The conceptual model generated as a concept map can be compared with the concept map representing the domain knowledge in order to identify misconceptions and help users to overcome them.
4. Finally, it could be discovered which are the incorrect paths in the concept map of the students and, try to correct them with the automatic generation of the natural language dialogue of the socratic tutor between the user and the system (more intuitive and easier to use than a more traditional application restricted to a set of fixed questions). The dialogue would end when the concept map of the student and the reference concept map coincide.

Appendix A: Engineering work

All the Will tools are on-line applications so that they can be freely accessed from any computer connected to Internet (through a web browser such as Firefox). They have been coded in Java Server Pages (JSP) and Javascript. The Apache Tomcat server has been used to interpret the JSP. Thus, it is also possible to run the systems locally in any machine with the Tomcat server installed. Regarding the data, it has been stored in a MySQL database to be accessed in SQL from the JSP applications with the driver JSP-MySQL. The simplified Entity-Relationship diagram of the database is depicted in Figure 1.

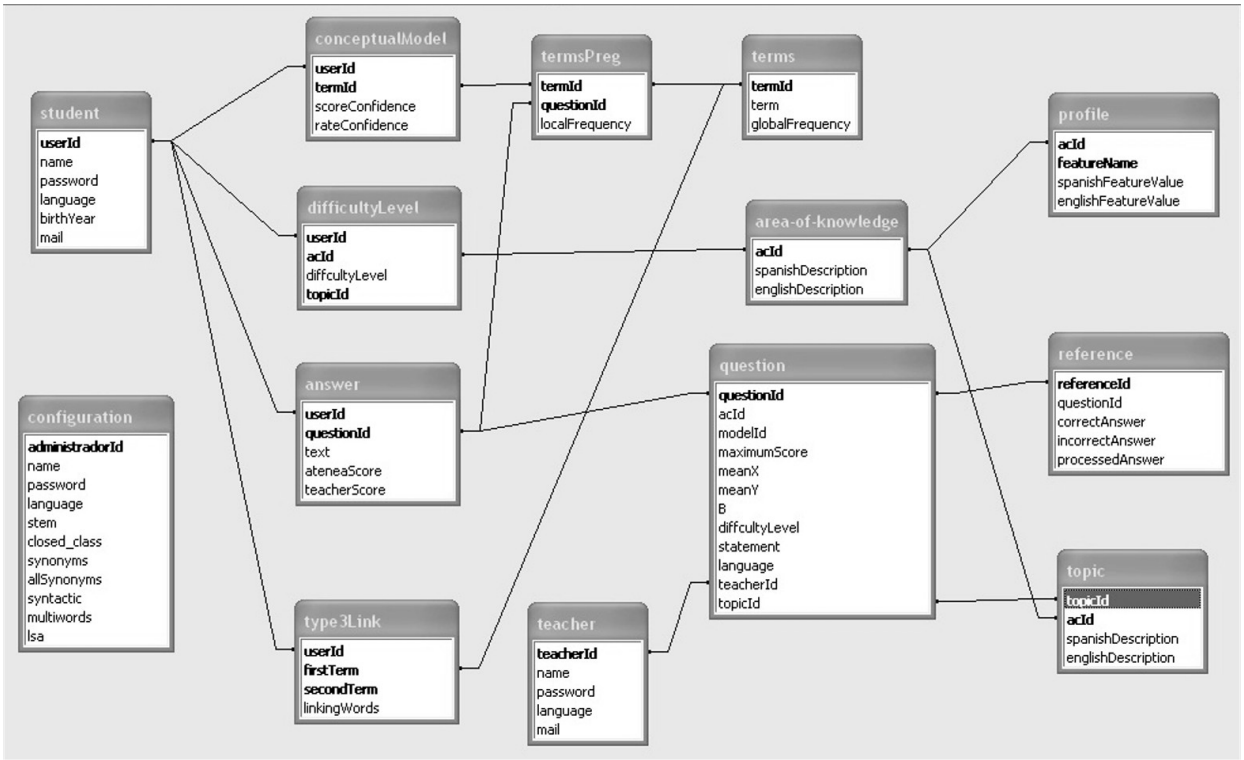


Figure 1: Simplified Entity-Relationship model of the Willow's database.

Currently, there is a complete collection of questions for an Operating System course. It consists of 20 questions extracted from exams of the Operating System subjects of Computer and Telecommunications Engineering degrees of the Universidad Autónoma de Madrid. They have been organized in five different topics with at least one question per level of difficulty in each topic. Per each question, at least three different references have been written by the teachers of the subjects. All of them have been introduced in the common database using Willed. The terms to be used as the BCs of the conceptual model were automatically extracted from these references.

Regarding the internal functionality of Willow, some programs have been written in C due to its optimum use of the resources (e.g. ERB with the genetic algorithms have been written in

a combination of C and Java) and, some scripts are in Perl due to its powerful parser of regular expressions (e.g. the scripts for using RARE are in Perl), whereas the rest of the code is in Java. The reason to choose Java is because it can be ported across operating systems without any modification. Besides, it is based on an object-oriented programming approach that facilitates the maintenance, reuse and scalability of the system. A brief outline of the evolution of Willow through versions is as follows:

- **The first version** was implemented in November 2003 after a few hours programming of the BLEU algorithm. It was modeled as a shell program without graphical interface and two parameters: the name of the file with the reference texts and the name with the file with the students answers.
- **The second version** was released in December 2003 and it added new statistical options such as calculating hypothesis contrast to find out the best n-gram scoring occurrence procedure and computing the linear regression values to transform the student scores in conformity with the teacher scale and not only between 0 and 1 as BLEU does. Therefore a third input parameter was incorporated to ERB: **c**, to calculate the correlation values achieved by the BLEU, the Keyword and the VSM procedures; **p**, to calculate the correlation attained with different number of reference texts; **h**, to calculate hypothesis contrasts; **r**, to calculate the linear regression between the teachers' scores and the system's ones; and **d**, to calculate the histogram with the distances between the teachers' scores and the system's ones.
- **The GA version** was released in January 2004 and it uses a genetic algorithm to find the upper bound of the ERB approach. It is based on the first version and it has not longer been reexamined since it already accomplishes its objective of finding out the best choice of reference texts.
- **The third version** was released in February 2004 and it introduced the MBP factor and the XML generation feature of the processed student answer. Below are an example of the resulting XML after having processed a student answer and the DTD employed. The main elements are n that stores the long of the n-gram found in the reference text, $nref$ that is the number of the reference where it has been found and $posC$ that indicates the position of this n-gram in the candidate text. The DTD used is:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!-- DTD for processing a student answer -->
<!ELEMENT answer (ngramFound+,ngramNoFound+,overlapFound+)>
<!ELEMENT ngramFound (#PCDATA)>
<!ATTLIST ngramFound nref CDATA "">
<!ATTLIST ngramFound n CDATA "">
<!ATTLIST ngramFound posC CDATA "">
<!ELEMENT ngramNoFound (#PCDATA)>
<!ATTLIST ngramNoFound n CDATA "">
<!ATTLIST ngramNoFound posC CDATA "">
<!ELEMENT overlapFound (#PCDATA)>
```

```
<!ATTLIST overlapFound nref1 CDATA "">
<!ATTLIST overlapFound nref2 CDATA "">
<!ATTLIST overlapFound n CDATA "">
<!ATTLIST overlapFound posC CDATA "">
```

An example of the XML generated is:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE answer SYSTEM "answer.dtd">
<answer>
<ngramFound nref="1" n="2" posC="0">means that </ngramFound>
<ngramNoFound n="1" posC="2">gives </ngramNoFound>
<ngramFound nref="1" n="1" posC="3">equal </ngramFound>
<ngramNoFound n="4" posC="4">the type of device </ngramNoFound>
<ngramFound nref="0" n="2" posC="8">with the </ngramFound>
<ngramNoFound n="2" posC="10">that interchange </ngramNoFound>
<ngramFound nref="1" n="2" posC="12">of data </ngramFound>
<ngramFound nref="0" n="3" posC="14">wants say that </ngramFound>
<ngramFound nref="1" n="1" posC="17">the </ngramFound>
<ngramNoFound n="5" posC="18">transference data speed </ngramNoFound>
<ngramFound nref="0" n="1" posC="23">no </ngramFound>
<ngramNoFound n="6" posC="24">changes as neither the type of </ngramNoFound>
<ngramFound nref="3" n="1" posC="30">transmission </ngramFound>
<ngramNoFound n="2" posC="31">(by flow </ngramNoFound>
<ngramFound nref="0" n="1" posC="33">or </ngramFound>
<ngramNoFound n="3" posC="34">by blocks) the </ngramNoFound>
<ngramFound nref="1" n="1" posC="37">independence </ngramFound>
<ngramFound nref="1" n="2" posC="38">of device </ngramFound>
<ngramNoFound n="4" posC="40">allow us effectively plan </ngramNoFound>
<ngramFound nref="3" n="3" posC="44">the input output </ngramFound>
<ngramFound nref="3" n="1" posC="47">be </ngramFound>
<ngramNoFound n="11" posC="48">this the type that is augmented as the efficiency
of processor </ngramNoFound>
</answer>
```

- **The first on-line version: Atenea** was released in March of the 2004 and it added a web-based interface to facilitate the use of the system. ERB was packaged as a Java Bean and the application programmed with Java Server Pages (JSP) with the Tomcat server.
- **The fourth version** was released in June of the 2004 and it included the wraetlic toolkit. ERB was combined with shallow NLP techniques to analyze which ones should be applied to achieve the higher correlation between Atenea's and the humans' scores. The on-line version was not modified.

- **The fifth version** was released in December of the 2004 and it provided the possibility of doing the comparison not only with ERB but also with LSA. In fact, the ITC-irst implementation of LSA was combined with ERB to analyze how it improved the accuracy of the scoring. The scripts in Perl to use RARE and the preprocessing necessary was also included in this version. Again all these changes did not modify the on-line version.
- **Willow**, as it is currently known, was released in October 2005. The processing and comparison module were fixed as the code of the fifth version. The JSP pages for the interface were improved with respect to the first on-line version to make the system more engaging and include all the features described. In fact, the main change with respect to Atenea is not in the scoring engine but in the use of AH techniques: the static were based on the use of a profile and the dynamic on the promotion-demotion of level procedure and the set of compensation questions. These AH techniques combined with the NLP makes possible to generate the students' conceptual model. Furthermore, Willed, Willow and COMOV were also implemented as the free-text ACAA authoring, configuring and visualizing tool. In this way, the system was ready to be used by the students.

The current parameters have been fixed by the administrator of Willow (although they are configurable and can be changed):

- Each question has three levels of difficulty (0, low; 1, medium; and, 2 high).
- There is a maximum limit of five features and five values per feature in the static component of each student's model.
- One hundred is the maximum number of scores chosen to do the calibration.
- The maximum length of the statement is 250 characters.
- The threshold age to be considered an adult is 16.

An example of fragment of a log generated by Willow for a student that participated in the third experiment (2006-2007 year) is (the answers are not reproduced here due to their long extension):

```
acId = Operating System |( topicId = Introduction )|all feedback ||no chronometer |10
questions |with previous feedback |angel background |arial |10 font size |black statement's
color |black student's answer color |green teacher answer's color | 80 text area width |10
text area height |
23/10/2006_21:3:40*1*An operating system is...*0.3*23/10/2006_21:7:3
23/10/2006_21:8:43*73*The multiprogramming degree is...*0.1*23/10/2006_21:14:26
23/10/2006_21:8:43*73*The multiprogramming degree is...*0.1*23/10/2006_21:16:23
23/10/2006_21:16:28*73*The processor is usually idle...*0.1*23/10/2006_21:21:44
23/10/2006_21:22:17*73*The memory has access...*0.0*23/10/2006_21:29:27
23/10/2006_21:31:24*73*Yes.*0.1*23/10/2006_21:31:57
23/10/2006_21:38:6*73*It is the number of processes that are in memory.*0.4*23/10/2006_21:38:44
23/10/2006_21:38:58*71*It is true as in the buddy algorithm the...*0.2*23/10/2006_21:54:0
23/10/2006_21:38:58*71*It is true this algorithm is...*0.4*23/10/2006_21:58:47
acId = Operating System | topicId = scheduling | 3 questions ||no chronometer | all feedback
| with previous feedback |angel background |arial |10 font size |black statement's color
```

```
|black student's answer color |green teacher answer's color |80 text area width |10 text
area height |
15/11/2006_1:8:52*61*Code in a process that...*0.5*15/11/2006_1:10:5
...
```

It can be seen how at the beginning of each session with Willow, it is indicated which area-of-knowledge and topic of questions have been chosen. Moreover, the amount of feedback, if previous feedback of failed questions should be given and the personalization options with their values. Next, for each question answered, there is a line in which it is indicated: the time in which the student saw the statement of the question for the first time, the identifier of the question, the answer provided, the score given by Willow and, finally the hour in which the student has pressed the button to continue with a different question.

The DTD created for the internal conceptual schema stored by Willow is:

```
   <?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT conceptual_model (terms, topics, area-of-knowledges, links)>
<!ATTLIST conceptual_model student ID #REQUIRED>
<!ELEMENT area-of-knowledges (area-of-knowledge+)>
<!ELEMENT area-of-knowledges (#PCDATA)>
<!ATTLIST area-of-knowledge id ID #REQUIRED>
<!ATTLIST area-of-knowledge state (white|blue|red|green|yellow) #REQUIRED>
<!ELEMENT topics (topic+)>
<!ELEMENT topic (#PCDATA)>
<!ATTLIST topic id ID #REQUIRED>
<!ATTLIST topic state (blue|red|green|yellow) #REQUIRED>
<!ELEMENT terms (term+)>
<!ELEMENT term (#PCDATA)>
<!ATTLIST term id ID #REQUIRED>
<!ATTLIST term state (green|red|yellow) #REQUIRED>
<!ATTLIST term CV CDATA "0.0">
<!ELEMENT links (link+)>
<!ELEMENT link (#PCDATA)>
<!ATTLIST link type (1|2|3) #REQUIRED>
<!ATTLIST link idTerm1 IDREF #REQUIRED>
<!ATTLIST link idTerm2 IDREF #REQUIRED>
<!ATTLIST link strong CDATA "0.0">
```

It can be seen the hierarchical structure of knowledge from area-of-knowledge, down to topic and term. Moreover, how the three types of links are considered and the information given to each link and term. An example of internal conceptual schema as generated in XML is:

```
   <?xml version="1.0" encoding="ISO-8859-1'"?>
<!DOCTYPE conceptual_model SYSTEM "model.dtd">
```

```
<conceptual_model student="12345">
<terms>
<term termId="bc1" state="red" CV="0.3">buddy algorithm</term>
<term termId="bc2" state="green" CV="0.9">Dekker algorithm</term>
<term termId="bc3" state="red" CV="0.3">readers-writers problem</term>
<term termId="bc4" state="red" CV="0.1">Peterson algorithm</term>
<term termId='bc5' state="yellow" CV="0.5">scheduling algorithm</term>
...
<term termIdid="bc64" state="red" CV="0.3">UNIX</term>
<term termId="bc65" state="red" CV="0.2">Windows</term>
</terms>
<topics>
<topic topicId="tc1" state="red">Introduction</topic>
<topic topicId="tc4" state="red">Scheduling</topic>
<topic topicId="tc3" state="red">Threads</topic>
<topic topicId="tc2" state="red">Processes</topic>
<topic topicId="tc5" state="red">Concurrency</topic>
</topics>
<area-of-knowledges>
<area-of-knowledge acId="ac1" state="red">Operating Systems </area-of-knowledge>
</area-of-knowledges>
<links> <link type="3" idTerm1="bc31" idTerm2="bc41" strong="0">is the number of</link>
<link type="3" idTerm1="bc42" idTerm2="bc20" strong="0">that must be executed by
</link>
...
<link type="2" idTerm1="tc1" idTerm2="bc15" strong="0.2">treats about</link>
<link type="2" idTerm1="tc2" idTerm2="bc10" strong="0.1">treats about</link>
<link type="2" idTerm1="tc2" idTerm2="bc40" strong="0.4">treats about</link>
<link type="2" idTerm1="tc5" idTerm2="bc7" strong="0.1">treats about</link>
...
<link type="1" idTerm1="tc1" idTerm2="ac1" strong="0.2">treats about</link>
<link type="1" idTerm1="tc2" idTerm2="ac1" strong="0.1">treats about</link>
<link type="1" idTerm1="tc3" idTerm2="ac1" strong="0.4">treats about</link>
<link type="1" idTerm1="tc4" idTerm2="ac1" strong="0.1">treats about</link>
<link type="1" idTerm1="tc5" idTerm2="ac1" strong="0.1">treats about</link>
</links>
</conceptual_model>
```

Finally, an example of log of use of COMOV is:

```
13/1/2007_14:56:52*The conceptual diagram of the student 12345 has been seen *13/1/2007_14:57:24
13/1/2007_14:56:22*The table ordered from high to low for the student 12345 has been seen
*13/1/2007_14:57:33
```

```
13/1/2007_14:58:28*The concept map of the whole class has been seen*13/1/2007_14:58:40
13/1/2007_14:58:40*The textual summary of the whole class has been seen*13/1/2007_15:00:36
```

Each line indicates that a representation format has been seen. Moreover, how long it has been reviewed as the line starts with the hour in which the button to see it was clicked and finishes with the hour in which the user returned to the main menu.

Appendix B: Data of the experiments

In this Appendix, the questionnaires used in the experiments described in Chapter 8 are gathered.

The first questionnaire about the usability of Willed consisted of the following ten items (five Likert-type, two multiple-choice and three free-text answers):

1. Rate the familiarity degree of using conceptual models for your classes.
2. Rate how intuitive you have found COMOV's interface.
3. Rate how informative you have found the table representation.
4. Rate how informative you have found the bars graph representation.
5. Rate how informative you have found the summary representation.
6. Rate how informative you have found the concept map representation.
7. In general, you prefer the table (T), the bars graph (B), the summary (S) or the concept map (C).
8. Please, justify your previous answer.
9. Rate how useful you find COMOV to see how well your students have understood the concepts of the lesson.
10. Would you use COMOV with your classes?
11. Please, justify your previous answer.
12. Would you recommend the use of COMOV to other courses' teachers?
13. Please, justify your previous answer.
14. Please, write any comment or suggestion that you have about COMOV or the idea of bridging the gap between what teachers think that students have understood and what they have actually understood.

The second questionnaire was completed by 32 students (16 used Atenea and 16 used Willow) to find out their satisfaction when using these free-text scoring systems. There was also Likert-type and free-text items:

1. Rate your familiarity with the on-line applications.
2. Rate your familiarity with the free-text scoring applications.
3. Rate how difficult you have found to use Atenea or Willow (depending on the system you have tried).
4. Rate how you feel that the system fits your needs.
5. Rate how intuitive you think that the interface is.
6. In general, rate how satisfied you are with the assessment session with Atenea or Willow.
7. Please, write how many questions you have answered:
8. Rate how adequate you think that the order of the questions is.
9. Rate how difficult you have found the questions asked.

Feature	**Willow**	**Atenea**	**Reason**	**Unknown**
Easier to use				
More adequate for my needs				
More intuitive				
More satisfactory to use				
A better order of the questions				
Easier questions				
Funnier				
More didactic				

10. Please, write how many questions you review Operating System per week.
11. If Atenea or Willow was freely available on the web, would you answer all questions dataset, only the sets of questions more difficult to understand or none?
12. Would you recommend the use of Atenea or Willow to a colleague in Operating System or other subject? Why?
13. Rate the following items about the feedback given by the system:
    - The automatic score.
    - The processed answer with the green color code.
    - The references.
    - The feedback of questions previously asked but failed.
14. How many references would you like to be shown?
15. Rate how fast you think that is the system processing each answer.
16. Please choose among the following features of the system the one that you think is the less interesting: all of them; the automatic scoring; the personalization options; the order of the questions; the end-of-session by time, number of questions or voluntary; other or none (i.e. all of them are interesting).
17. Please choose among the following features of the system the one that you think is the most interesting: all of them; the automatic scoring; the personalization options; the order of the questions; the end-of-session by time, number of questions or voluntary; other or none.
18. I would improve the system by...
19. I would like that the system in order to assess me would take into account...

The third questionnaire is a comparison questionnaire between Atenea and Willow for the students of the second experiment (2005-2006 year) that were willing to use both systems during a week from any computer connected to Internet having full control over the systems' features.

1. Choose the type of access to Internet that you have at home: no Internet, modem, ADSL, optic fiber or other.
2. I prefer to use Atenea or Willow from: my house because..., the laboratory at the university because... or other.
3. Rate how useful you think that the promotion-demotion difficulty level procedure of Willow is.
4. Regarding the order of the questions in Willow:
    - I like that it goes from easy to more difficult questions.

- I prefer that it goes from difficult to easy questions.
- I prefer that it starts with medium difficulty questions.
- I prefer to choose the difficulty level of the next question.
- Other:....

5. Once used Willow and Atenea, I think that for the following features, the system I tick is better because of the reason I also provide (when unknown is marked is because I do not have an opinion about it):
6. Indicate which system (Atenea or Willow) you have used more times.
7. Write how you think about these systems once you have had enough time to test them.
8. Indicate, in general, which system you prefer (Atenea or Willow).
9. Specify how long have you used the system.
10. Indicate which order of questions is your favorite.
11. Comment if you think that your answer has been correctly processed with the systems.
12. Write a brief diary in which day per day you write your notes/comments/impressions about the use with Atenea or Willow.

The questionnaire about COMOV is already depicted in Table 8.3. Finally, the 2006-2007 questionnaires are very similar to the previous year ones. The main differences are that all comparison items are removed and only the questions focused on Willow remain; besides, the diary item is also removed as it was not very well accepted by the students the last year (too much work); and, new questions about COMOV are introduced.

Appendix C: Introduction (in Spanish)

Muchos filósofos, psicólogos e investigadores han examinado la naturaleza del conocimiento, sus posibles clasificaciones y representaciones. Entre todas las definiciones de conocimiento que se pueden encontrar en la literatura, quisiera resaltar la de Kang and Byun [2001], puesto que ha sido una de las fuentes de inspiración de este trabajo: "*El conocimiento es el producto de una actividad de aprendizaje según la cual un individuo asimila y enlaza la nueva información en su estructura cognitiva conforme a cómo comprende su entorno.*".

Esto es, en este trabajo, se sigue un enfoque constructivista, según el cual, construimos nuestro propio conocimiento a partir de las experiencias e interacciones que tenemos con el mundo [Carpendale, 1997]. De hecho, cada estudiante construye su propia estructura cognitiva o modelo conceptual (entendido como una red de conceptos) dependiendo de sus características individuales y conocimientos previos. Por lo tanto, la forma en que cada estudiante entiende los nuevos conceptos es distinta. Incluso, según la teoría del Aprendizaje Significativo de Ausubel et al. [1978], podría darse el caso de que el estudiante no pudiera aprender los nuevos conceptos debido a la falta de conceptos previos con los que enlazarlos en su modelo conceptual.

Las formas tradicionales de evaluación basadas en el uso de tests no son capaces de identificar muchos de los errores conceptuales de estos estudiantes, encontrándose ejemplos en la literatura de estudiantes capaces de obtener buenas notas en dichos tests y que sin embargo, al ser evaluados mediante un conjunto de preguntas abiertas en una entrevista personal, han reflejado graves carencias conceptuales [Sigel, 1999].

Por lo tanto, es necesario tener alguna estrategia fiable, para modelar la estructura conceptual de cada estudiante. En este trabajo, se propone un nuevo procedimiento capaz de generar los modelos conceptuales de los estudiantes de una forma completamente automática a partir de sus respuestas en texto libre. Además, se propone que dichos modelos conceptuales puedan ser mostrados tanto a los profesores como a los estudiantes tantas veces como ellos quieran durante el curso. El procedimiento se basa en la combinación sinérgica de técnicas de Procesamiento de Lenguaje Natural (PLN) e Hipermedia Adaptativa (HA), y se ha implementado en las **Herramientas Will** que constan de (todos los sistemas están libremente disponibles on-line para su uso académico):

- **Willow**, un sistema de evaluación automática y adaptativa de respuestas en texto libre [http2].
- **Willed**, una herramienta de autor [http3].
- **Willoc**, una herramienta de configuración [http4].
- **COMOV**, un visualizador de modelos conceptuales [http5].

Objetivos

El principal objetivo de este trabajo es **construir un puente entre lo que los estudiantes realmente aprenden en clase y lo que los profesores piensan que han enseñado**. La

motivación no es simplemente poner automáticamente una nota a los estudiantes, o darles más retroalimentación, sino proporcionar a los profesores la posibilidad de tener acceso inmediato al modelo conceptual de cada estudiante de forma específica, y de toda la clase en general (lo que es muy difícil de conseguir a mano).

Este objetivo engloba los siguientes subobjetivos:

- **Para profesores:**
  - Tener retroalimentación que les informe de hasta qué punto los estudiantes están comprendiendo los conceptos enseñados y les permita identificar sus errores conceptuales más comunes.
  - Proporcionarles la posibilidad de seguir la evolución conceptual de cada estudiante y de toda la clase. Esto es, de averiguar cómo los nuevos conceptos están modificando los previos y qué enlaces se están creando entre ellos.
- **Para estudiantes:**
  - Proporcionarles una forma de evaluación nueva que se adapte a su ritmo de estudio y les permita escribir respuestas en texto libre a preguntas abiertas y recibir de forma inmediata retroalimentación sobre cómo han respondido.
  - Proporcionarles la representación gráfica del modelo conceptual generado a partir de las respuestas que han introducido en el sistema de evaluación de respuestas de texto libre, para que puedan identificar ellos mismos sus principales errores conceptuales y compararlos con los del resto de la clase.
  - Fomentar el pensamiento reflexivo y crítico de los estudiantes, evitando mostrarles las respuestas correctas siempre que fallen la pregunta, y en su lugar, guiarles hacia las respuestas correctas mediante un conjunto de preguntas orientativas automáticamente generadas por el sistema adaptativo de evaluación de respuestas en texto libre.
- **Otros:**
  - Superar las limitaciones impuestas tradicionalmente en la evaluación de cursos proporcionados por sistemas de educación a distancia, permitiendo a los estudiantes responder preguntas abiertas.
  - Adaptar la evaluación de respuestas en texto libre, de forma que se base en el modelo de estudiante generado (que a su vez es modificado por la evaluación), usando técnicas tanto de Procesamiento de Lenguaje Natural como de Hipermedia Adaptativa.
  - Encontrar la mejor combinación de técnicas de Procesamiento de Lenguaje Natural capaces de mejorar la evaluación de respuestas en texto libre, sin necesidad de entrenamiento ni de pedir a los profesores rellenar plantillas de evaluación.

Ejemplo de escenario

Para ilustrar los objetivos previamente mencionados en un ejemplo práctico, supongamos que tenemos un grupo de estudiantes matriculados en la asignatura de Sistemas Operativos de la carrera de Ingeniería de Telecomunicaciones. Además, que en esta asignatura, se sigue un

sistema de evaluación continua que obliga a los estudiantes a asistir a las clases, puesto que la nota final que obtendrán no sólamente será la del examen final sino también considerará, en un cierto porcentaje, el trabajo que han realizado durante el cuatrimestre.

El profesor de la asignatura, Juan, se enfrenta a la tarea de enseñar un temario bastante complejo puesto que todos los temas están bastante interrelacionados y existen grandes dependencias conceptuales entre ellos. Juan necesitaría saber si los estudiantes han comprendido los conceptos iniciales para poder avanzar con conceptos más complejos. Por lo tanto, él siempre antes de empezar con algo nuevo, pregunta si se ha comprendiendo lo anterior. Sin embargo, ningún estudiante le responde. A Juan también le gustaría poder hacer más exámenes y ejercicios prácticos durante el cuatrimestre pero no tiene tiempo de corregirlos ni de hacerlos en clase si quiere terminar con el temario.

Este escenario es el caso ideal para usar un sistema automático y adaptativo de respuestas en texto libre. Puesto que es un sistema on-line, ningún estudiante lo tiene que instalar. De hecho, no se necesita ningún conocimiento informático para usarlo (a parte de saber usar un navegador web). Juan simplemente tiene que introducir en el sistema el conjunto de preguntas que quiere hacer a sus estudiantes y las respuestas que él considera correctas para estas preguntas usando una herramienta de autor que ha sido desarrollada para hacer esta tarea más sencilla (también es una aplicación web que no necesita ningún conocimiento informático).

A continuación, debería incluir como una de las actividades de la evaluación continua el uso de Willow unas horas por semana. Los estudiantes de Juan pueden acceder al sistema en los laboratorios de la facultad o desde su casa. El único requisito es que el ordenador tenga acceso a Internet.

Todos los estudiantes de Juan reciben una cuenta en el sistema cuando completan el proceso de registro (que también es on-line). En el registro, se les pregunta su nombre, edad e idioma (necesario para su modelo). El nombre y la password proporcionados los pueden usar para acceder al sistema desde cualquier lugar en cualquier momento. Para cada sesión particular, pueden elegir cuántas preguntas quieren contestar, cuánto tiempo quieren practicar, los temas e incluso el nivel de retroalimentación (sólo la nota, la nota con la respuesta procesada, la nota con las respuesta procesada y las respuestas correctas proporcionadas por Juan), y luego empiezan a contestar las preguntas del sistema, según su modelo que a su vez se está actualizando a medida que los alumnos van respondiendo más preguntas. Además, un sistema de logs está registrando todas sus acciones.

Cuando un estudiante falla una pregunta, en lugar de darle las respuestas correctas, se le guía hacia la respuesta correcta mediante un conjunto de preguntas de orientación. Finalmente, cuando el estudiante pasa la pregunta o suspende todas las preguntas de orientación, entonces se le proporciona la retroalimentación que haya seleccionado. En todo caso, si la pregunta no ha sido superada, se marca para ser preguntada de nuevo a ese estudiante.

De esta forma, los estudiantes de Juan se pueden beneficiar de retroalimentación inmediata y reforzar los puntos que tienen más flojos (evaluación formativa). Los estudiantes saben que cuánto más practiquen, más oportunidades tienen de aprobar el exámen final. Además, pueden organizar su estudio de forma que sea más efectivo y concentrarse en los conceptos previos que

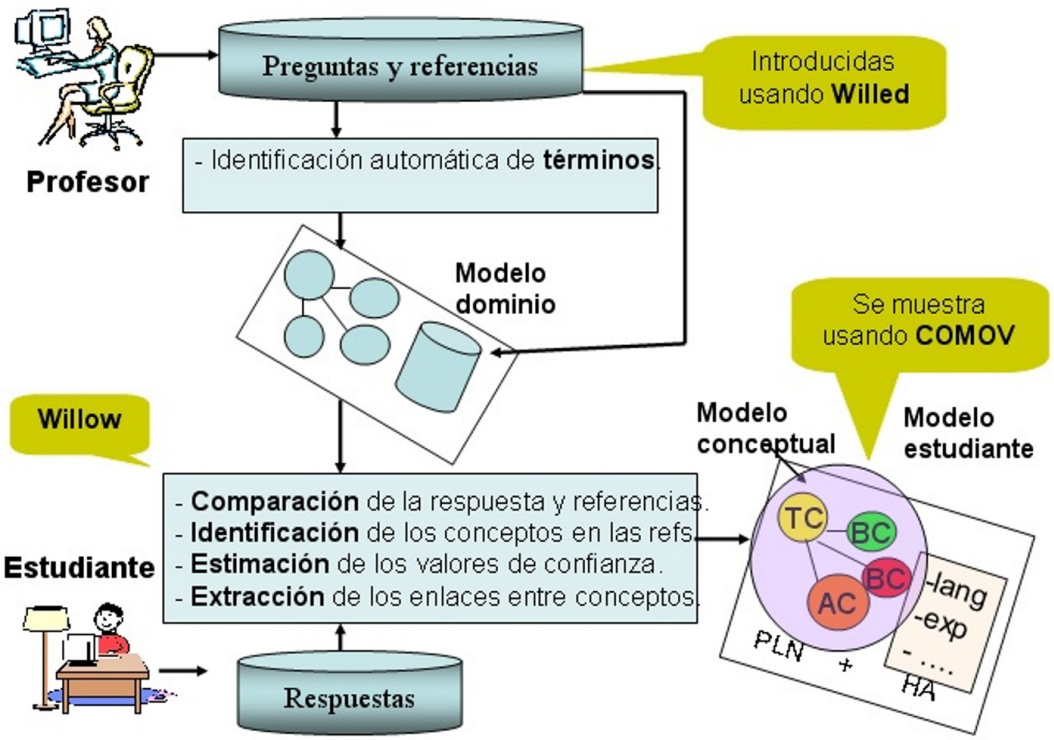


Figure 2: Representación gráfica del procedimiento descrito en este trabajo.

les faltan antes de tratar de aprender conceptos más complejos.

Por otro lado, Juan también consigue cumplir su objetivo de ser capaz de proporcionar más ejercicios a sus estudiantes sin necesidad de tener que corregirlo él ni de ocupar tiempo de clase. Además de conseguir más retroalimentación sobre los estudiantes, puesto que tiene la posibilidad de visualizar el modelo conceptual de cada estudiante y el de toda la clase mediante COMOV. De esta forma, puede ver qué conceptos son los que a los estudiantes les está costando más comprender, y sus errores conceptuales. Por lo tanto, Juan tiene la retroalimentación que necesita para ajustar el temario y evitar empezar temas con conceptos demasiado complejos y que los estudiantes aún no van a ser capaces de comprender, puesto que les faltan los conceptos previos necesarios.

Vista global la tesis

Idea general La Imagen 2 proporciona una vista global del procedimiento propuesto para generar automáticamente los modelos conceptuales de los estudiantes. Como se puede comprobar, se pide al profesor que use una herramienta de autor para introducir las preguntas y sus respuestas correctas (referencias) en la base de datos. A continuacón, las referencias son automáticamente procesadas para generar el modelo de dominio.

A medida que los estudiantes van respondiendo las preguntas planteadas por el sistema de evaluación automática y adaptativa de respuestas en texto libre, no sólamente consiguen retroalimentación inmediata sino también el sistema analiza el uso de los términos para ir

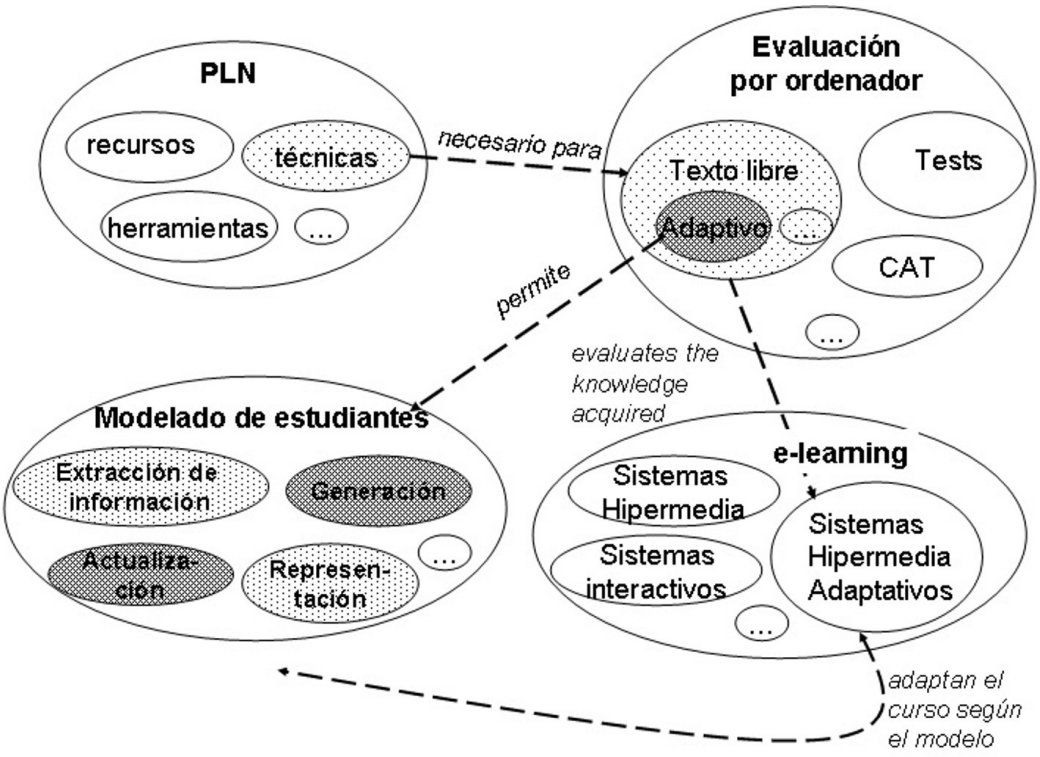


Figure 3: Campos relacionados con este trabajo y cómo se enmarca este trabajo entre ellos
.

generando su modelo conceptual. De esta forma, el modelo de estudiante consiste no sólo de información personal proporcionada por el estudiante sino también del modelo conceptual generado.

Finalmente, el modelo conceptual se puede mostrar a los profesores y a los estudiantes con un visualizador de modelos conceptuales para que puedan identificar qué conceptos deberían revisarse todavía más y, cuáles han sido ya comprendidos.

Contexto Es importante destacar que debido a la naturaleza interdisciplinar de los objetivos que se persiguen, este trabajo no está solamente relacionado con un campo, sino con varios como se muestra en la Imagen 3.

En particular, está relacionado con:

- **Evaluación por ordenador** puesto que este campo estudia cómo se pueden usar los ordenadores para poder evaluar el aprendizaje de los estudiantes.
- **Modelado de estudiantes** puesto que este campo estudia cómo modelar los estudiantes de forma que la información obtenida de los modelos se puede usar como retroalimentación para profesores, estudiantes o ser usada internamente por el sistema.
- **Hipermedia Adaptativa Educativa** puesto que este campo estudia las técnicas para adaptar el curso a impartir según el modelo de cada estudiante.
- **Procesamiento de Lenguaje Natural** puesto que este campo estudia las técnicas para procesar texto libre, y por lo tanto, entre ellas las que sirven para evaluar automáticamente

respuestas en texto libre.

Organización Este trabajo se organiza en tres grandes apartados:

- **Revisión del estado del arte**: Comprende los capítulos 2-4 en los que se presentan las bases teóricas de este trabajo y se revisa el estado del arte de los principales campos a los que ha hecho alguna contribución:
  - *El capítulo 2* describe algunas teorías cognitivas y pedagógicas en las que se basa este trabajo. Esto es, teorías que validan que se puede inferir un modelo mental, de hecho, un modelo conceptual representable en varios formatos, como por ejemplo mapas conceptuales.
  - *El capítulo 3* define lo que es el modelado de estudiantes, y en particular, el modelado conceptual de estudiantes. Además también revisa algunas técnicas para construir estos modelos y sistemas que se basan en ellos.
  - *El capítulo 4* proporciona una revisión del estado del arte de evaluación de respuestas en texto libre: técnicas estadísticas, de Procesamiento de Lenguaje Natural y otras que se están usando actualmente, y cómo se implementan en los sistemas existentes en el campo.
- **Propuesta**: Los capítulos del 5 al 7 describen el procedimiento para generar automáticamente los modelos conceptuales de los estudiantes, y los sistemas que implementan dicho procedimiento.
  - *El capítulo 5* detalla el procedimiento de generación de los modelos conceptuales dando un ejemplo, basado en el escenario previamente descrito, y que explica paso a paso el procedimiento propuesto.
  - *El capítulo 6* se centra en el sistema de evaluación automática y adaptativa de respuestas en texto libre llamado Willow. Se describe su arquitectura a alto nivel, y cada uno de los módulos que componen esta arquitectura. La herramienta de autor Willed y la de configuración Willoc también son descritas en este capítulo.
  - *El capítulo 7* introduce el visualizador de modelos conceptuales implementado llamado COMOV. Su objetivo es mostrar tanto a los estudiantes como a los profesores los modelos conceptuales generados en cinco formas de representación distintas: mapa conceptual, diagrama conceptual, tabla, gráfica de barras y resumen textual.
- **Experimentos y conclusiones**: Los capítulos 8 y 9 cierran el documento con la explicación de los experimentos realizados, sus resultados, las conclusiones obtenidas y las líneas de trabajo futuro.
  - *El capítulo 8* valida la viabilidad del procedimiento de generación de los modelos conceptuales desde el punto de vista de los profesores y los estudiantes. Además, describe el diseño y los resultados de los experimentos realizados que prueban cómo se han cumplido los objetivos expuestos.
  - *El capítulo 9* resume las principales conclusiones de este trabajo y cómo se puede extender. También proporciona la información necesaria para aplicar el procedimiento en otro idioma y/o área de conocimiento.

Finalmente, este trabajo tiene cinco apéndices: el Apéndice A presenta los detalles técnicos de la implementación de los sistemas descritos en el texto; el Apéndice B da información extra sobre los experimentos; el Apéndice C es la traducción española del Capítulo 1; el Apéndice D es la traducción española del Capítulo 9; y, el Apéndice E proporciona los términos en español tal y como fueron usados en los experimentos realizados con los estudiantes españoles (recuérdese que en el texto principal del documento se presentan sus versiones traducidas a inglés para facilitar la lectura a las personas interesadas en este documento pero que no sepan español).

Appendix D: Conclusions (in Spanish)

Los profesores quieren que los estudiantes adquieran ciertos conceptos y sus relaciones como están normalmente aceptados en el contexto de un curso. Sin embargo, lo que los profesores intentan transmitir a los estudiantes y lo que realmente los estudiantes comprenden es, en algunas ocasiones, bastante distinto [Sigel, 1999]. Una posible razón que explica esta situación se puede encontrar en la teoría constructivista del aprendizaje, según la cual cada estudiante construye su propia estructura cognitiva dependiendo de su conocimiento previo. Esto es, cada estudiante tiene su propio modelo conceptual, que se puede definir como la red de conceptos y relaciones entre estos conceptos.

Por lo tanto y, según la Teoría del Aprendizaje Significativo de Ausubel [Ausubel, 1968], cada estudiante sólo será capaz de comprender los conceptos nuevos si tienen algún tipo de relación con los conceptos previos de su modelo conceptual. Sin embargo, las formas tradicionales de evaluación basadas en tests no son capaces de detectar el modelo conceptual de cada estudiante y por lo tanto, fallan en detectar sus errores conceptuales [Sigel, 1999].

En las últimas décadas, se han creado formas más sofisticadas de evaluación formativa como evaluar respuestas en texto libre [Valenti et al., 2003] o, enganchar a los estudiantes en actividades de repaso para diagnosticar su falta de conocimiento [Dimitrova, 2003]. De todas formas, ninguno de los sistemas actuales de evaluación de respuestas en texto libre tiene en cuenta ningún tipo de modelo de estudiante y, ninguno de los sistemas capaces de trabajar con modelos de estudiantes, puede extraer información a partir de las respuestas de los estudiantes en texto libre para mejorar los modelos.

En este trabajo, se propone un procedimiento para generar de forma automática los modelos conceptuales de cada estudiante y de toda la clase a partir de las respuestas proporcionadas al sistema de evaluación automático y adaptativo. Estos sistemas son la evolución de los sistemas actuales de evaluación de respuestas en texto libre, de forma que no sólo evaluan respuestas en texto libre automáticamente sino que lo de hacen de forma adaptada al modelo de cada estudiante. Este procedimiento se ha implementado en las herramientas Will que constan de: Willow, un sistema de evaluación de respuestas en texto libre automático y adaptativo; Willed, una herramienta de autor; Willoc, una herramienta de configuración; y, COMOV, un visualizador de modelos conceptuales.

Se han realizado varias demostraciones de las herramientas Will en conferencias internacionales como la "*Vigésimo segunda conferencia internacional de la Sociedad Española de Procesamiento de Lenguaje Natural*" y workshops especializados como el "*Workshop internacional en aplicaciones de la web semántica para e-learning*". Además, las herramientas Will se han probado en varios experimentos que se han llevado a cabo en la asignatura de Sistemas Operativos de la carrera de Ingeniería de Telecomunicaciones de la Universidad Autónoma de Madrid en los cursos 2005-2006 y 2006-2007.

Estos experimentos han servido para probar la viabilidad del procedimiento de construcción de los modelos conceptuales que sirve de puente entre lo que los profesores intentan enseñar y lo que los estudiantes realmente comprenden (tal y como se refleja en los modelos conceptuales generados). En particular, en las próximas secciones se verá como se han cumplido los objetivos propuestos para profesores, estudiantes y el resto de subobjetivos indicados en el Capítulo 1. Además, también se explicará cómo se puede extender el procedimiento a otros dominios y/o idiomas y se destacará las nuevas características que distinguen las herramientas Will del resto de sistemas relacionados. Finalmente, se terminará este apéndice con las líneas que quedan abiertas como trabajo futuro.

Objetivos cumplidos para los profesores

En el Capítulo 1 se destacaron dos objetivos principales para los profesores: proporcionarles más retroalimentación sobre el grado de asimilación que tienen sus estudiantes de los conceptos expuestos en las clases y sus principales errores conceptuales, y darles la oportunidad de seguir la evolución conceptual de cada estudiante y de toda la clase.

Respecto al primer objetivo, se ha cumplido puesto que los profesores pueden ver con COMOV, en cualquier momento, la representación que ellos decidan (mapa conceptual, diagrama conceptual, tabla, gráfica de barras o resumen textual) del modelo conceptual de los estudiantes (de un solo estudiante o de toda la clase) tal y como ha sido generado por Willow. Además, se ha proporcionado a los profesores una taxonomía de errores que pueden detectar a partir de los modelos generados, como se describe en la Sección 5.4.6

Respecto al segundo objetivo, también se ha cumplido puesto que los profesores pueden seguir el progreso de sus estudiantes accediendo en varias ocasiones consecutivas a COMOV durante el curso. Esto es, por ejemplo, si ven el mapa conceptual al principio del curso, luego un mes después y luego dos meses después, y los comparan, los profesores podrán observar cómo se han ido modificando los valores de confianza de los conceptos y cómo se han ido creando nuevos enlaces entre los conceptos.

Finalmente, de los experimentos que se han realizado, también se puede concluir que:

- Los profesores creen que COMOV es bastante útil y usable. Todos han afirmado que lo usarían en sus clases para tener información acerca de lo bien que los estudiantes han comprendido los conceptos expuestos en sus clases. Además, los profesores recomiendan el uso de las Will tools a sus compañeros en otras asignaturas.
- Los profesores han dado una puntuación muy buena (entre 3 y 5 en una escala de 0 a 5) a todas las representaciones del modelo conceptual implementadas. Esto podría ser porque todas son complementarias y se centran en destacar distintos aspectos del modelo conceptual. En todo caso, cuando se ha preguntado a los profesores de forma directa qué forma de representación les parece que refleja mejor la estructura conceptual de los estudiantes, los mapas conceptuales han tenido más votos. Este hecho apoya la afirmación de Novak de que los mapas conceptuales son una de las herramientas más poderosas para visualizar el conocimiento conceptual [Novak and Gowin, 1984].

- Los profesores han validado los modelos conceptuales tanto cuantitativamente como cualitativamente. Cualitativamente, puesto que han tenido la oportunidad de comprobar cómo los estudiantes que obtenían una puntuación más alta en el examen, tenían un mapa conceptual más complejo que aquellos estudiantes que obtenían una puntuación más baja. Cuantitativamente, puesto que se ha encontrado una correlación estadísticamente significativa (50%, p=0.0068) entre las puntuaciones proporcionadas por los estudiantes en el examen final y el valor promedio de los valores de confianza de los conceptos de los modelos conceptuales generados para dichos estudiantes.
- Los profesores han resaltado la importancia de recibir más retroalimentación y la posibilidad de seguir la evolución conceptual de sus estudiantes (esto es, la posibilidad de ir viendo cómo los estudiantes van asimilando los nuevos conceptos).

Objetivos cumplidos para los estudiantes

En el Capítulo 1, se indicaron tres objetivos a cumplir para los estudiantes: proporcionarles un procedimiento capaz de evaluar sus respuestas en texto libre de forma automática y adaptativa, ayudarles a identificar sus principales errores conceptuales, y guiarles hacia la respuesta correcta en lugar de directamente mostrársela para fomentar el pensamiento crítico y reflexivo.

Respecto al primer objetivo, se ha cumplido puesto que se ha dado acceso a los estudiantes a una nueva herramienta de evaluación formativa: el sistema de evaluación automática y adaptativa Willow. Willow proporciona como retroalimentación no sólo la nota, sino también la respuesta procesada con un código de colores para indicar los puntos fuertes y débiles de cada respuesta (esta información se les recuerda la próxima vez que se les presenta esta pregunta, en caso de que no lograran superarla la primera vez que se les planteó) y las respuestas correctas de los profesores. De hecho, debido a la gran cantidad de retroalimentación que Willow puede dar a los estudiantes, se ha permitido que sean ellos mismos los que decidan qué quieren recibir cómo retroalimentación (sólo la nota; la nota con la respuesta procesada; la nota, la respuesta procesada y las respuestas correctas de los profesosres). Además, se ha probado que Willow cumple la mayoría de los requisitos indicados en el estudio de Darus et al. [2001], como se explica en el Capítulo 6.

Respecto al segundo objetivo, también se ha cumplido puesto que los estudiantes tienen también acceso a COMOV y por lo tanto, a ver el modelo conceptual que se ha ido generando mientras usaban Willow. De esta forma, ellos mismos pueden identificar los conceptos que ignoran (y tienen un valor de confianza muy bajo) y qué enlaces entre conceptos les faltan (no están en su modelo pero si en el modelo de toda la clase).

Respecto al tercer objetivo, también se ha cumplido puesto que se ha implementado en Willow un sistema de preguntas de orientación hacia la respuesta. De esta forma, siempre que un estudiante falla una pregunta, entra en el diálogo de orientación hacia la respuesta correcta. En la primera pregunta, se le da la posibilidad al estudiante de dar una explicación más extensa a la pregunta. Si con la respuesta proporcionada, continúa suspendiendo, entonces se le presenta una segunda pregunta de orientación acerca de un concepto relacionado con el tema y que está

marcado en el modelo conceptual como desconocido. En el caso de que tampoco supere la pregunta, se le hace una tercera pregunta del tipo Sí/No con justificación. Sólamente cuando el estudiante falla también esta tercera pregunta, se le presentan las respuestas correctas y la pregunta se marca como suspendida para ser preguntada en una sesión posterior. Nótese cómo estas preguntas de orientación ayudan a los estudiantes a construir la respuesta correcta, puesto que en el caso de los estudiantes del tercer experimento, el 74.14% de las preguntas que consiguieron superar, fueron superadas gracias al uso del conjunto de estas preguntas de orientación hacia la respuesta.

Finalmente, a partir de los tres experimentos que se han llevado a cabo con Willow, también se puede concluir que:

- Los estudiantes encuentran útil y fácil de usar Willow, destacando que les ayuda a reforzar conceptos.
- Los estudiantes aprecian tener disponibles las respuestas correctas de los profesores para saber cómo deberían responder las preguntas.
- Aunque ninguno de los profesores les dijo, en ningún momento, que usar Willow fuera divertido, los estudiantes resaltaron que repasar usando Willow era interactivo y que les resultaba divertido. Además, dijeron que se habían sentido motivados para seguir contestando más preguntas que si sólo hubieran tenido que leer del libro de la asignatura y hacer los ejercicios a mano.
- Los estudiantes prefieren la versión adaptativa del sistema (Willow) a la versión no-adaptativa (Atenea). Esto es porque, a pesar de que en el primer experimento con los estudiantes la diferencia entre ambos sistemas no fuera estadísticamente significativa (nótese que sólo pudieron usar el sistema durante veinte minutos y con todas las opciones prefijadas), en los experimentos siguientes cuando los estudiantes pudieron usar los sistemas sin ningún tipo de restricción, Willow fue mayoritariamente usada. De hecho, los estudiantes dijeron que Willow se ajusta mejor a sus necesidades, que el orden de las preguntas es más adecuado, y en general, que se sienten más satisfechos con este sistema. Además, los estudiantes que usaron Willow fueron capaces de responder más preguntas y consiguieron una nota ligeramente superior en el post-test.
- Sobre las opciones de personalización, aunque en el primer y segundo experimento, ningún estudiante cambió ninguno de los valores por defecto, en el tercer experimento, cuando tenían más información sobre las características de Willow y el curso completo (desde Octubre 2006 hasta Enero 2007) para usarlo, se observó que sí las usaban y cambiaban los valores por defecto.
- En el tercer experimento, los estudiantes también han podido validar los modelos conceptuales. Han podido seguir su evolución conceptual durante el cuatrimestre, y han apreciado positivamente la retroalimentación proporcionada.

Subobjetivos cumplidos

En el Capítulo 1, se han indicado tres subobjetivos principales: superar las limitaciones impuestas por los tests de las secciones típicas de evaluación de los sistemas de educación a distancia, adaptar la evaluación de las respuestas en texto libre, y encontrar la combinación óptima de técnicas de Procesamiento de Lenguaje Natural para mejorar la evaluación automática de respuestas en texto libre.

Respecto al primer subobjetivo, se ha cumplido puesto que este trabajo se ha centrado en la posibilidad de evaluar respuestas en texto libre. Esto se puede hacer con sistemas que funcionen de forma autónoma o integrados con sistemas Hipermedia Adaptativos Educativos como se explica en la Sección 6.1

Respecto al segundo subobjetivo, se ha propuesta la evolución de los sistemas de evaluación de respuestas en texto libre, de forma que tengan en cuenta el modelo de estudiante. La idea clave continúa siendo la misma, esto es, cuanto más similar es la respuesta del estudiante a las referencias proporcionadas por los profesores, mejor se puede considerar que es dicha respuesta y por lo tanto, mayor la nota que obtenga. La mejora está en que ahora:

- Se mantiene un modelo de estudiante que contiene dos grandes componentes: uno estático y otro dinámico. En el componente estático, se guardan los valores de las características del curso tal y como las determina el profesor en la herramienta de autor. De esta forma, el curso se presenta a los estudiantes adaptado a estos valores. En el componente dinámico, se guardan los valores que cambian a medida que los estudiantes responden preguntas del sistema de evaluación de respuestas automático y adaptativo.
- Se implanta un procedimiento nuevo de cambio de nivel de dificultad de las preguntas planteadas en función del modelo del estudiante, y un conjunto de preguntas de orientación que se generan a partir de las deficiencias encontradas en el modelo para guiar a los estudiantes hacia la respuesta correcta.

Respecto al tercer subobjetivo, de las técnicas de Procesamiento del Lenguaje Natural que se han estudiado, la combinación óptima para español es ERB y lematización, alcanzado una correlación de Pearson entre las notas de los profesores y de Willow para el mismo conjunto de preguntas de un 54%; y, para inglés de un 56% usando ERB, lematización, eliminación de palabras funcionales y Análisis de Semántica Oculta (ambos son resultados del estado del arte). Finalmente, a partir de los experimentos realizados, también se puede concluir que:

- Sólo usar ERB no es suficiente para construir un sistema de evaluación de respuestas en texto libre. Sin embargo, ERB se puede usar como un módulo de comparación léxica. De hecho, ERB ha mejorado otras técnicas estadísticas relacionadas como palabras clave, Modelo del Espacio Vectorial o Análisis de Semántica Oculta que se están usando en sistemas como E-rater [Burstein et al., 1998] o IEA [Foltz et al., 1999].
- El principal punto débil de ERB, esto es su dependencia en la calidad de las referencias, se puede aliviar usando técnicas de Resolución de Anáforas para generar nuevas referencias a partir de las ya existentes. Además usando Algoritmos Genéticos se puede escoger automáticamente el conjunto de referencias para el próximo año a partir no sólo de las

referencias de este curso sino también de las mejores respuestas proporcionadas por los estudiantes. De hecho, usando este procedimiento se ha mejorado la correlación promedio hasta un 63% en los textos españoles en los que se ha probado (alcanzándose hasta un 70% de correlación en preguntas del tipo ventajas/desventajas y Sí/No con justificación, que como se ha visto suelen ser las más difíciles de evaluar para Willow).

- Los resultados son válidos includo usando versiones traducidas al inglés, puesto que la traducción no ha producido una gran variabilidad en el vocabulario, como se demostró en el Capítulo 8.
- Las técnicas de Resolución de Anáforas no son útiles para mejorar los resultados de métricas de similitud de co-ocurrencias de n-gramas como ERB, al menos cuando se usan textos traducidos automáticamente de español a inglés.
- Se ha probado el buen funcionamiento del enfoque del módulo de Identificación de Términos como una tarea de clasificación basada en el algoritmo C4.5 de Quinlan, alcanzando una F-score de 0.74 tanto para español como para inglés. Además, el valor de la cobertura (recall) es superior al de la precisión (precision), lo que es apropiado puesto que la lista de términos extraídos pueden ser posteriormente revisada por los profesores.
- Se pueden extraer términos incluso a partir de corpus pequeños de dominios muy específicos (sólo unas cuatro mil palabras). Esto podría deberse al hecho de que las referencias escritas por los profesores son normalmente de muy alta calidad, y por lo tanto sólamente a partir de ellas, ya se puede obtener suficiente información para extraer los términos.
- Cuánto mayor es el tamaño del corpus para el entrenamiento de Análisis de Semántica Oculta, mejores son los resultados. De hecho, es mejor recoger un corpus más grande, aunque sea más genérico, que uno más específico en el dominio bajo estudio.
- En el caso del inglés, idioma del que se tenían todos los recursos para probar la combinación de ERB con Análisis de Semántica Oculta, se encontró que esta combinación obtiene siempre mejores resultados que usar cada método de forma independiente. Además cuando LSA no está disponible y no se hace entrenamiento, los resultados sólo se reducen en un 2%, 54% entre las puntuaciones del profesor y del sistema.

Comparación con sistemas relacionados

En primer lugar, nótese que todas las referencias de los sistemas que se citan en esta sección se proporcionan en la revisión del estado del arte en los Capítulos 3 y 4 y por lo tanto, no se vuelven a repetir aquí. Además, nótese también que debido a la falta de una lista de características relevantes que debieran cumplir todos estos sistemas, a partir de la lista de características mencionadas en los sistemas revisados del estado del arte, se ha cofeccionado la lista que se muestra en la Tabla 4. La Tabla 5 muestra los sistemas qué cumplen cada una de estas características.

Como se puede comprobar, las herramientas Will ofrecen algunas características deseables que no están presentes en otros sistemas, como: la capacidad de generar modelos conceptuales de individuos o grupos; proporcionar evaluación automática y adaptativa de respuestas

Id	Característica
1	Soportar el aprendizaje significativo
2	Fomentar el pensamiento crítico y reflexivo
3	Mostrar la evolución de los modelos conceptuales de los estudiantes
4	Ofrecer una herramienta de autor que cubra todos los aspectos del curso
5	Evaluar la organización de los conceptos
6	Dar feedback instántaneo a los profesores
7	Proporcionar apoyo emocional
8	Usar un modelo conceptual inspeccionable de cada individuo y de toda la clase
9	Tener soporte para varios idiomas
10	Generar los modelos conceptuales individuales y del grupo de forma automática
11	Proporcionar evaluación automática y adaptativa de las respuestas en texto libre
12	Usar las referencias en texto libre sin necesidad de plantillas
13	Eliminar la necesidad de entrenamiento para evaluar las respuestas en texto libre
14	Permitir la personalización de la interfaz
15	Mostrar los modelos conceptuales de los estudiantes en varios formatos

Table 4: Lista de características.

en texto libre (sin necesidad de entrenamiento y con las referencias en texto libre); permitir la personalización de la interfaz; y, mostrar el modelo conceptual de cada estudiante en varios formatos de representación. Los sistemas más relacionados a las herramientas Will son STyLE-OLM+GISMO [Mazza and Milani, 2004] y E-TESTER [Guetl et al., 2005]. Todos ellos persiguen un objetivo similar: proporcionar más feedback a los profesores de forma que puedan identificar los principales errores conceptuales de sus estudiantes y, ayudarles a organizar sus conceptos.

Sin embargo, Willow difiere de STyLE-OLM en un punto muy importante: Willow genera automáticamente el modelo conceptual a partir de las respuestas de los estudiantes en texto libre, mientras que en STyLE-OLM el modelo se negocia entre el sistema y el estudiante. STyLE-OLM se basa en entablar un diálogo interactivo entre el estudiante y el sistema mediante el uso de grafos conceptuales para ir construyendo de esta forma el modelo del estudiante (sin usar ninguna técnica de Procesamiento de Lenguaje Natural). El modelo del estudiante es una proyección del modelo del dominio que se ha introducido previamente como una ontología en el sistema y se ha guardado en XML. El modelo así generado se puede mostrar con CourseVis como una matriz cognitiva y con GISMO como un conjunto de dependencias entre los conceptos o histogramas. No se generan mapas conceptuales, diagramas conceptuales, tablas, gráfico de barras o resumenes textuales.

La principal mejora de las herramientas Will con respecto a E-TESTER es que no sólamente tienen en cuenta los conceptos, sino también las relaciones entre ellos (esto es, se genera un modelo conceptual completo), mientras que E-TESTER sólo muestra un histograma de la frecuencias de uso de los términos por parte del estudiante en su respuesta y del profesor en la suya. Además, la característica de E-TESTER de generar preguntas del tipo "*Qué es XXX*", también está incluida en Willow. En particular, son el segundo tipo de las preguntas de orientación, y Willow no sólamente genera este tipo de preguntas, sino también las que piden una explicación más general de la respuesta (primer tipo) y las de Sí/No a partir de la infomación de

Id	Sistemas
1	ALE, COMPASS, ConceptLab, CREEK-Tutor, E-TESTER, KBS Hyperbook, LEO, StyLE-OLM, **Willow**
2	ALE, ConceptLab, CREEK-Tutor, DynMap+, KBS Hyperbook, StyLE-OLM, **Willow**
3	ALE, ConceptLab+VisMod, DynMap+, LEO, STyLE-OLM+GISMO, **Willow+COMOV**
4	ALE, COMPASS, DynMap+, E-TESTER, KBS Hyperbook, LEO, TADV, **Willow**
5	AEA, COMPASS, ConceptLab, C-rater, Jess, MarkIT, MRW, RMT, **Willow**
6	ALE, VisMod, DynMap+, E-TESTER, GISMO, TADV, **Willow**
7	KBS Hyperbook, RMT, STyLE-OLM, VisMod, **Willow**
8	ConceptLab, DynMap+, **Willow**
9	IntelliMetric, **Willow**
10	**Willow+COMOV**
11	**Willow**
12	**Willow**
13	**Willow**
14	**Willow**
15	**COMOV**

Table 5: Sistemas de los revisados en los capítulos 3 y 4 que cumplen las características identificadas en la Tabla 4.

las referencias (tercer tipo), con el propósito de guiar al estudiante hacia la respuesta correcta.

El resto de los sistemas revisados están relacionados únicamente de forma parcial con las herramientas Will. Todos tienen en común que usan algún tipo de modelo conceptual, pero sus objetivos son distintos. Por ejemplo, ALE está más orientado a permitir a los estudiantes navegar en el espacio semántico interactivo a través de mapas conceptuales que no son generados por el sistema, sino introducidos usando la herramienta de autor WINDS. COMPASS, ConceptLab+VisMod, KBS Hyperbook y LEO tienen como objetivo enseñar nuevos conceptos a los estudiantes. En relación a TADV, se centra en producir consejos únicamente para los profesores (pero no para los estudiantes).

Por último, es importante destacar que todos los sistemas que se revisaron en el Capítulo 4 se centran exclusivamente en la evaluación automática de respuestas en texto libre, sin tener en cuenta ningún modelo de estudiante o usar ninguna técnica estática o dinámica de hipermedia adaptativa. Esto puede ser un problema para evaluación formativa ya que toda la información sobre los estudiantes (nivel de conocimiento, preferencias, etc.) no se tiene en cuenta y por lo tanto, la mayoría de los estudiantes no encuentran suficiente motivación para continuar contestando preguntas.

Extendiendo el procedimiento a otro idioma y/o dominio

Los ejemplos que se proporcionan en este trabajo se han tomado del área de conocimiento de Sistemas Operativos, dado que yo trabajo como profesora en este área. Sin embargo, el procedimiento descrito se puede aplicar a otras áreas de conocimiento si se siguen los pasos que se explican en esta Sección, siempre y cuando no sea necesario evaluar pensamiento creativo o cálculos matemáticos, lo que está fuera del alcance de este trabajo. Además, también se puede aplicar a otros idiomas distintos al español y al inglés, simplemente teniendo en cuenta los requisitos de la Table 6. Los pasos a dar son los siguientes:

Técnica PLN	**Español**	**Inglés**	**Otro**
ERB*	Sí	Sí	Sí
Lematización*	Sí	Sí	Sí
Eliminación de palabras funcionales*	Sí	Sí	No
Identificación de Términos*	Sí	Sí	No
Análisis de Semántica Oculta	No	Sí	No
Algoritmos Genéticos	Sí	Sí	Sí
Resolución de Anáforas	No	Sí	No

Table 6: Técnicas de Procesamiento de Lenguaje Natural (PLN) que se pueden usar para el procedimiento de generación de los modelos conceptuales (las técnicas que son obligatorias están marcadas con un asterisco) y, los idiomas en los que está actualmente disponibles en Willow.

1. Si el idioma en el que se quiere aplicar el procedimiento es español o inglés, entonces los profesores y los estudiantes pueden usar los sistemas implementados sin ninguna modificación adicional. Esto es, puesto que las interfaces de todos los sistemas se han diseñado en ambos idiomas. En el caso de otros idiomas, es necesario traducir las interfaces.
2. A continuación, los profesores deben solicitar una cuenta para acceder a Willed. Esto lo pueden hacer enviando un correo al administrador de Willed, tal y como aparece en la página de registro.
3. Una vez los profesores tienen ya su cuenta de acceso, tienen que crear la nueva área de conocimiento con Willed y rellenar todos los campos que se le solicitan: el nombre, la descripción, características y temas del área.
4. El nombre del curso se guarda en la base de datos de Willow como el concepto de área de conocimiento (AC).
5. El nombre de los temas del curso se guardan en la base de datos de Willow como los conceptos de tema (TCs). Se crean los enlaces tipo 1 entre el AC y cada TC.
6. Los profesores introducen las preguntas usando Willed. En particular, para cada pregunta es necesario: su enunciado y las referencias en tantas versiones como valores para las características a las que se quiera adaptar el curso, la puntuación máxima, el tema al que pertenece y el nivel de dificultad.
7. Con respecto a las referencias y debido a su importancia, es aconsejable que al menos haya tres referencias distintas por pregunta. Además, deben estar escritas por profesores distintos para capturar el máximo número posible de formas distintas de expresar el mismo contenido.
8. Los profesores ingleses también pueden pedir al administrador de Willed que genere nuevas referencias a partir de las tecleadas usando Resolución de Anáforas. En el caso del resto de idiomas, esta característica no está todavía disponible, porque RARE necesita un modelo para cada idioma, y actualmente sólo está desarrollado para inglés.
9. Los profesores ingleses y españoles no necesitan proporcionar ninguna otra técnica o recurso de Procesamiento de Lenguaje Natural, y pueden ir directamente al siguiente paso. Mientras que para otro idioma diferente, sería necesario tener un lematizador, un identificador del tipo de palabra, y un corpus específico y genérico para el módulo de Identifi-

cación de Términos. Es necesario clasificar los n-gramas de las referencias como términos o no términos. El corpus específico se puede construir a partir de las propias referencias introducidas en Willed y el corpus genérico se puede recuperar automáticamente de la web.

10. El administrador de Willow debería aplicar el módulo de Identificación de Términos a las referencias proporcionadas por los profesores para generar una lista de términos. Esta lista puede ser revisada por los profesores. La lista de términos resultante se guarda en la base de datos de Willow como conceptos base (BCs) del modelo conceptual junto con la frecuencia con la que aparecen en las referencias de los profesores. Se crean los enlaces de tipo 2 entre cada BC y el TC al que pertenece.
11. Los profesores dicen a sus alumnos que ya se pueden registrar en Willow enviando un mail al administrador de Willow.
12. Los estudiantes responden las preguntas que están guardadas en la base de datos usando Willow, que no sólamente evalúa las preguntas sino también analiza el uso que el estudiante está haciendo de los conceptos base para asignar a cada uno de ellos el valor de confianza que le corresponda según las métricas definidas en el Capítulo 5. Además, Willow busca patrones BC+verbo+BC para ir creando los enlaces tipo 3 entre los conceptos base de cada modelo conceptual.
13. Durante el curso, los profesores pueden ver el modelo conceptual de cada estudiante o de un grupo de estudiantes usando COMOV. Los estudiantes también se pueden registrar en COMOV siempre que quieran ver su propio modelo conceptual de estudiante y el modelo conceptual de toda la clase.
14. Para el próximo curso, los profesores que deseen usar Willow de nuevo, pueden pedir al administrador de Willow que ajuste algunos parámetros internos para mejorar la evaluación de las respuestas en texto libre para el próximo año. En particular, a partir de la información de este curso, se puede:
    - Ejecutar el algoritmo genético en las referencias y las respuestas de los estudiantes para elegir cuál es el mejor conjunto de referencias a partir de todos estos textos.
    - Calibrar las líneas de regresión para escalar la puntuación interna de Willow de 0 a 1, a la escala proporcionada por el profesor para cada pregunta.
    - Pedir a los profesores que evalúen a mano un conjunto de preguntas para calcular la correlación de Pearson entre las puntuaciones automáticas y las del profesor para estas respuestas. Esto es porque la combinación óptima de técnicas para otras áreas de conocimiento y/o idiomas podría ser diferente.

Trabajo futuro

Se planea continuar mejorando el procedimiento de generación de los modelos conceptuales mediante:

- La aplicación del procedimiento a otras áreas de conocimiento y/o idiomas distintos de Sistemas Operativos y español.

- La exploración de posibilidades más sofisticadas de generar el modelo conceptual del grupo.
- La creación de plantillas tipo de buenos y malos mapas conceptuales, para que los profesores puedan reconocerlos de un vistazo.
- La introducción de la posibilidad de proporcionar acceso al contenido relacionado del curso mediante cliquear en los nodos (conceptos) o enlaces (relaciones entre conceptos) del mapa conceptual, para que Willow pueda también cubrir las deficiencias encontradas.
- La nueva posibilidad de permitir a los profesores y a los estudiantes modificar el modelo.
- La mejora del conjunto de preguntas de orientación. Por ejemplo, aplicando técnicas de generación de lenguaje natural para crear nuevas preguntas abiertas según la evaluación que ha hecho Willow del modelo conceptual. De esta forma, Willow podrá no sólamente evaluar las respuestas en texto libre, sino también generar nuevas preguntas específicas para cada estudiante según su particular nivel de conocimiento y características de su modelo. Se espera que esta línea de trabajo culmine con el desarrollo de un tutor socrático capaz de de promover el aprendizaje significativo entablando un diálogo con los estudiantes basado en el modelo conceptual.

Como se puede comprobar, la intención es continuar explotando las posibilidades de la combinación de las técnicas de Hipermedia Adaptativa y de Procesamiento de Lenguaje Natural y mejorando las herramientas Will a partir de los comentarios de profesores y estudiantes. En particular:

- Intentando esquemas de combinación entre las notas de ERB y Análisis de Semántica Oculta y probándolos tanto para español como para inglés.
- Mejorando el parser sintáctico para español e inglés.
- Implementando el módulo español para RARE.
- Integrando nuevas técnicas como Extracción de Información y Análisis Retórico en el Módulo de Procesamiento de Willow. Se espera que las técnicas de Extracción de Información sean útiles puesto que se han usado con éxito en otros sistemas de evaluación de respuestas en texto libre. Por otro lado, el Análisis Retórico puede ayudar a identificar tanto en la respuesta del estudiante como en las referencias, los fragmentos en los que se citan ventajas o desventajas, donde se proporciona una conclusión, etc. Además, las técnicas de Extracción de Información pueden ayudar a mejorar las extracción de los enlaces de tipo 3 entre los conceptos base.
- Incluyendo más adaptación dinámica en Willow, como actualizando el nivel de dificultad de cada pregunta según las respuestas proporcionadas por los estudiantes o permitiendo que los estudiantes se muevan entre las preguntas libremente, con un código de color que les indique si la pregunta es de su nivel o no.

Finalmente, se plantea la posibilidad de generalizar el procedimiento de generación de los modelos conceptuales no sólamente para estudiantes, sino en general para cualquier usuario y en particular, para personas mayores o gente con algún tipo de discapacidad mental. Como se ha visto, Willow ya tiene varias características de personalización que permiten a los usuarios cambiar partes de la interfaz para hacerla más simple para ellos: el tipo de fuente se puede aumentar para gente mayor y, algunos elementos de la interfaz se pueden quitar para personas

con algún tipo de discapacidad mental.

En general, para los usuarios que no pueden usar Willow (por ejemplo, porque no son capaces de usar un navegador web), sería posible permitirles escribir libremente. Posteriormente, el texto en formato electrónico se procesaría con Willow, resultando en un proceso completamente transparente para los usuarios, que únicamente verían el resultado final: el modelo conceptual generado y representado de forma adaptada a cada uno de ellos. Para conseguir este objetivo, las siguientes subtareas se tienen que realizar:

1. El texto se procesa con un lematizador, eliminador de palabras funcionales, Módulo de Identificación de Términos, de Reconocimiento de Entidades Nominales y Extracción de Información para detectar y relacionar entidades como personas, lugares o fechas.
2. La salida de esta primera subtarea es una lista con los principales términos (conceptos), entidades y sus relaciones que se han usado en el texto. A partir de esta información, el modelo conceptual se puede generar y ser representado de una forma gráfica.
3. El modelo conceptual representado como un mapa conceptual se puede comparar con un mapa conceptual representando el modelo del dominio, para identificar errores conceptuales y ayudar a los usuarios a superarlos.
4. Se podría descubrir las rutas incorrectas en el mapa conceptual de cada estudiante e, intentar corregirlas mediante la generación automática de un diálogo en lenguaje natural en forma de tutor socrático entre el usuario y el sistema (más intuitivo y fácil de usar que una aplicación tradicional restringida a un conjunto fijo de preguntas). De esta forma, el diálogo terminará cuando el mapa conceptual de cada estudiante y el mapa conceptual de referencia coincidan.

Appendix E: Examples (in Spanish)

In this Appendix, the Spanish terms for the representation formats of the conceptual model shown in Chapter 7 are presented together with their manual translations into English. The terms have been grouped according to the type of concept that they label: area-of-knowledge concept (AC), topic concepts (TCs) and basic concepts (BCs). They are listed in alphabetical order (indexed by the English term).

The AC in English is: **Operating Systems** that is the translation of the Spanish term: Sistemas Operativos.

The TCs are:

Introduction Introducción

Concurrency Concurrencia

Processes Procesos

Scheduling Planificación

Threads Hilos

The BCs are:

active process proceso activo

active waiting espera activa

addresses space espacio de direcciones

answer time tiempo de respuesta

atomization atomización

batch processing procesamiento por lotes

batch system sistema batch

buddy algorithm algoritmo de colegas

clock interruption interrupción reloj

concurrent execution ejecución concurrente

concurrent reading lectura concurrente

concurrent writing escritura concurrente

context space espacio de contexto

control block bloque de control

control register registro de control

critical section sección crítica

cyclic scheduling planificación cíclica

data area área de datos

deadlock interbloqueo

Dekker algorithm algoritmo de Dekker

dispatcher planificador

father process proceso padre

FCFS FCFS

HRRN HRRN

input output entrada salida

kernel stack pila del núcleo

KLT KLT

Linux Linux

main memory memoria principal

memory block bloque de memoria

memory partition partición de memoria

monoprocessor system sistema monoprocesador

multiprogramming multiprogramación

multiuser system sistema multiusuario

mutual exclusion exclusión mutua

operating system sistema operativo

Peterson algorithm algoritmo de Peterson

priority process prioridad del proceso

process image imagen del proceso

process pid pid del proceso

process queue cola del proceso

process proceso

processor register registro del procesador

quantum quanto

reader writer lector escritor

register registro

resource recurso

returning time tiempo de retorno

rotatory turn turno rotatorio

round robin scheduling planificación round robin

running stack pila de ejecución

safe sequence secuencia segura

safe state estado seguro

scheduling algorithm algoritmo de planificación

semaphore semáforo

shared memory memoria compartida

starvation inanición

state register registro de estado

stay time tiempo de estancia

thread hilo

time slice rodaja de tiempo

timeout timeout

UNIX UNIX

user stack pila de usuario

Windows Windows

References

S. Abney. *Part-of-speech tagging and partial parsing.* Dordrecht: Kluwer, 1996.

V. Aleven, B. McLaren, I. Roll, and K. Koedinger. Toward tutoring help seeking: Applying cognitive modeling to meta-cognitive skills. In *Intelligent Tutoring Systems*, Lecture Notes in Computer Science. Springer Verlag, 2004.

E. Alfonseca. *An Approach for Automatic Generation of on-line Information Systems based on the Integration of Natural Language Processing and Adaptive Hypermedia techniques.* PhD thesis, Computer Science Department, Universidad Autónoma de Madrid, 2003.

E. Alfonseca and D. Pérez. Automatic assessment of short questions with a BLEU-inspired algorithm and shallow NLP. In *Advances in Natural Language Processing*, volume 3230 of *Lecture Notes in Computer Science*, pages 25–35. Springer Verlag, 2004b.

E. Alfonseca, D. Pérez, and P. Rodríguez. Automatic multilingual generation of on-line information sites. In *Proceedings of the second International Conference on Multimedia and Information Technologies for the Education (MICTE)*, 2003.

E. Alfonseca, R.M. Carro, M. Freire, A. Ortigosa, D. Pérez, and P. Rodríguez. Educational Adaptive Hypermedia meets Computer Assisted Assessment. In *Proceedings of the International Workshop of Educational Adaptive Hypermedia, collocated with the Adaptive Hypermedia (AH) Conference*, 2004a.

E. Alfonseca, R.M. Carro, M. Freire, A. Ortigosa, D. Pérez, and P. Rodríguez. Authoring of Adaptive Computer Assisted Assessment of Free-text Answers. *Educational Technology and Society (ETS), Special Issue on Authoring of Adaptive Hypermedia*, 8(3), 2005.

E. Alfonseca, A. Moreno-Sandoval, J.M. Guirao, and M. Ruiz-Casado. The wraetlic NLP suite. In *Proceedings of the Language and Resources Conference Evaluation (LREC)*, 2006.

J. Allan, T. Allan, and N. Sherkat. Automated assessment of children's handwritten sentence responses. In *Proceedings of the 7th International Computer Assisted Assessment Conference*, 2003.

D. Appelt and D. Israel. Introduction to Information Extraction technology. IJCAI 99 Tutorial, 1999.

D.P. Ausubel. *The Psychology of meaningful verbal learning.* New York: Grune and Stratton, 1963.

D.P. Ausubel. *Educational psychology: A cognitive view.* New York: Holt, Rinehart and Winston, 1968.

D.P. Ausubel, J.D. Novak, and H. Hanesian. *Educational Psychology: a cognitive view, 2nd. ed.* Holt, Reinhart and Winston, New York, 1978.

P. Baffes and R.J. Mooney. A novel application of theory refinement to student modeling. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, volume 403408, 1995.

A. Ballester, A. Martín-Municio, F. Pardos, J. Porta-Zamorano, R.J. Ruiz-Urena, and F. Sánchez-León. Combining statistics on n-grams for automatic term recognition. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, 2002.

T. Barnes. The Q-matrix Method: Mining Student Response Data for Knowledge. *American Association for Artificial Intelligence 2005 Workshop on Educational Datamining*, 2005.

T. Barnes and D. Bitzer. Fault tolerant teaching and automated knowledge assessment. *Proceedings of the ACM*, 2002.

M. Berry. Large-scale sparse singular value computations. *International Journal of Supercomputer Applications*, 6(1):13–49, 1992.

J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Kluwer Academic Publishers Norwell, MA, USA, 1981.

M. Birenbaum, K. Tatsuoka, and Y. Gutvirtz. Effects of response format on diagnostic assessment of scholastic achievement. *Applied psychological measurement*, 16(4), 1992.

P. Bishop. Assessment for a purpose. *MSOR Connections*, 2(3), 2002.

D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

B.S. Bloom. *Taxonomy of educational objectives: The classification of educational goals.* Handbook I, cognitive domain. New York ; Toronto: Longmans, Green, 1956.

D. Bourigault. Surface grammatical analysis for the extraction of terminological noun phrases. In *Proceedings of COLING*, pages 977–981, Nates, France, 1992.

J.S. Bruner, J.J. Goodnow, and G.A. Austin. *A study of thinking.* New York, NY: Wiley, 1956.

S. Bull and T. Nghiem. Helping learners to understand themselves with a learner model open to students, peers and instructors. In *Proceedings of Workshop on Individual and Group Modelling Methods that Help Learners Understand Themselves, International Conference on Intelligent Tutoring Systems*, volume 2002, pages 5–13, 2002.

S. Bull, A.T. McEvoy, and R. Eileen. Learner models to promote reflection in combined desktop pc/mobile intelligent learning environments. In *Proceedings of the Learner Modelling for Reflection Workshop, AIED2003*, 2003.

H.L. Burns and C.G. Capps. Foundations of Intelligent Tutoring Systems: An Introduction. *Foundations of Intelligent Tutoring Systems*, pages 1–19, 1988.

J. Burstein. The E-rater scoring engine: Automated essay scoring with natural language processing. *Automated Essay Scoring: A Cross-Disciplinary Perspective. Lawrence Erlbaum, Mahwah, NJ*, 2003.

J. Burstein and M. Chodorow. Automated scoring for non-native english speakers. In *Proceedings of the ACL99 Workshop on Computer-Mediated Language Assessment and Evaluation of Natural Language Processin, College Park, MD*, 1999.

J. Burstein, K. Kukich, S. Wolff, C. Lu, M. Chodorow, L. Bradenharder, and M. Dee Harris. Automated scoring using a hybrid feature identification technique. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics*, 1998.

J. Burstein, C. Leacock, and R. Swartz. Automated evaluation of essays and short answers. In *Proceedings of the 5th International CAA Conference*, 2001.

A.J. Caas, D.B. Leake, and D.C. Wilson. Managing, mapping and manipulating conceptual knowledge. In *AAAI Workshop Technical Report WS-99-10: Exploring the Synergies of Knowledge Management and Case-Based Reasoning*, California, U.S.A, 1999. AAAI Press.

A.J. Caas, J.W. Coffey, M.J. Carnot, P. Feltovich, J. Feltovich, R.R. Hoffman, and J. Novak. A summary of literature pertaining to the use of concept mapping techniques and technologies for education and performance support. Technical report, Naval Education and Training, Florida, U.S.A., 2003.

M.T. Cabré, R. Estopá, and J. Vivaldi. Automatic term detection: a review of current systems. *Recent advances in computational terminology*, 2:53–87, 2001.

J.P. Callan, W.B. Croft, and J. Broglio. TREC and TIPSTER experiments with INQUERY. Information Processing and Management, 1995.

D. Callear, J. Jerrams-Smith, and V. Soh. CAA of Short Non-MCQ answers. In *Proccedings of the 5th International Computer Assissted Assessment conference*, 2001.

A. Carlson and S. Tanimoto. Text Classification Rule Induction in the Presence of Domain-Specific Expression Forms. Mixed Language Explanations in Learning Environments. In *Proceedings of the Artifical Intelligence in Education (AIED) conference*, 2005.

J.I.M. Carpendale. An explanation of Piaget's constructivism: Implications for social cognitive development. *The development of social cognition*, pages 36–64, 1997.

R.M. Carro, E. Pulido, and P. Rodríguez. TANGOW: Task-based Adaptive learNer Guidance On the Web. In *Proceedings of the Second Workshop on Adaptive Systems and User Modeling on the Web in the Eight International World Wide Web Conference*, 1999.

M. Catt and G. Hirst. An intelligent CALL system for grammatical error diagnosis. *Computer Assisted Language Learning*, 3:3–27, 1990.

CDCP. Ebola hemorrhagic fever: Table showing known cases and outbreaks, in chronological order, 2002. http://www.cdc.gov/ncidod/dvrd/spb/mnpages/dispages/ebotabl.htm.

H. Cen, K.R. Koedinger, and B. Junker. *Learning Factor Analysis - A General Method for Cognitive Model Evaluation and Improvement*, pages 164–175. Springer-Verlag, 2006.

H. H. Chen, S. Y. Cheng, and J. S. Heh. Assessing users' mental knowledge by using structural approach and concept map. *Proceedings of 2005 International Conference on Machine Learning and Cybernetics, vols 1-9*, pages 2166–2171, 2005.

J. R. Christie. Automated essay marking - for both style and content. In *Proceedings of the 3rd International Computer Assisted Assessment Conference*, 1999.

J.R. Christie. Automated essay marking for content - does it work? In *Proceedings of the 7th International Computer Assisted Assessment Conference*, 2003.

G.K. Chung and H.F. O'Neill. Methodological approaches to online scoring of essays. Technical Report 461, UCLA, National Center for Research on Evaluation, Student Standards, and Testing, 1997.

J. W. Coffey. LEO: A concept map based course visualization tool for instructors and students. *Knowledge and information visualization: searching for synergies*, 3426:285–301, 2005.

J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

K. Cotton. *Monitoring Student Learning in the Classroom.* Northwest Regional Educational Laboratory, 1988.

J. Cowie and W.G. Lehnert. Information Extraction. *Communications of the ACM*, 39(1): 80–91, 1996.

D. Cristea and G.E. Dima. An integrating framework for anaphora resolution. *Information Science and Technology*, 4(3), 2001.

A. Cucchiarelli, E. Faggioli, and P. Velardi. Will very large corpora play for semantic disambiguation the role that massive computing power is playing for other AI-hard problems? In *Proceedings of the 2nd. Conference on Language Resources and Evaluation (LREC)*, 2000.

S. Darus and S.H. Stapa. Lecturers' expectations of a computer-based essay marking systems. *Journal of the Malaysian English Language Teachers' Association (MELTA)*, pages 47–56, 2001.

S. Darus, S. Hussin, and S.H. Stapa. Students' expectations of a computer-based essay marking system. In Jayakaran Mukundan, editor, *Reflections, visions and dreams of practice: Selected papers from the IEC 2001 International Education Conference*, pages 197–204, 2001.

A. Datar, N. Doddapaneni, S. Khanna, V. Kodali, and A. Yadav. EGAL - Essay Grading and Analysis Logic, 2004.

S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6): 391–407, 1990.

P. Dessus, B. Lemaire, and A. Vernier. Free text assessment in a virtual campus. In *Proceedings of the 3rd International Conference on Human System Learning*, pages 61–75, 2000.

D. Dicheva and C. Dichev. Educational topic maps. In *Proceedings of ISWC*, Hiroshima, Japan, 2004.

R.J. Dietel, J.L. Herman, and R.A. Knuth. What does research say about assessment. *North Central Regional Educational Laboratory, Oak Brook*, 1991.

V. Dimitrova. Interactive cognitive modelling agents–potential and challenges. In *Proceedings of 6th International Conference ITS 2002 Workshop, Spain*, pages 52–62, 2002.

V. Dimitrova. STyLE-OLM: Interactive Open Learner Modelling. *International Journal of Artificial Intelligence in Education*, 13(1):35–78, 2003.

P. Drouin. Term extraction using non-technical corpora as a point of leverage. *Terminology*, 9 (1), 2003.

P. Eklund and R. Wille. A multimodal approach to term extraction using a rhetorical structure theory tagger and formal concept analys. In *Proceedings of Second International Conference on Co-operative Multimodal Communication: Theory and Applications*, pages 171–175, Tilburg, Netherlands, 1998.

R. Florian, S. Cucerzan, C. Schafer, and D. Yarowsky. Combining classifiers for Word Sense Disambiguation. *Natural Language Engineering*, 8(4):327–341, 2002.

P. Foltz, D. Laham, and T. Landauer. The Intelligent Essay Assessor: Applications to Educational Technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 1(2), 1999.

T. Foltz, W. Kintsch, and T. Landauer. The Measurement of Textual Coherence with Latent Semantic Analysis. *Discourse Processes*, 25(2-3), 1998. Special Issue: Quantitative Approaches to Semantic Knowledge Representations.

M. Freire and P. Rodríguez. A graph-based interface to complex hypermedia structure visualization. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI), ACM Press*, pages 163–166, 2004.

E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie. Recent soft computing approaches to user modeling in adaptive hypermedia. *Adaptive Hypermedia and adaptive web-based systems, Proceedings of 3rd Int Conf Adaptive Hypermedia-AH*, pages 104–113, 2004.

J.M. Galusha. Barriers to learning in distance education. *Interpersonal Computing and Technology*, 5(3):6–14, 1997.

H. Geller. Concept mapping, e-learning and science education. on-line: http://www.physics.gmu.edu/ hgeller/GWUelearn, 2004.

H. Giouroglou and A.A. Economides. State-of-the-art and adaptive open-closed items in adaptive foreign language assessment. In *Proceedings of the 4th Hellenic Conference with International Participation: Informational and Communication Technologies in Education*, pages 747–756, 2004.

D.H. Gitomer, L.S. Steinberg, and R.J. Mislevy. *Diagnostic assessment of troubleshooting skill in an Intelligent Tutoring System*, pages 73–101. Hillsdale, NJ: Lawrence Erlbaum Associates, 1995.

R. Glasser, A. Lesgold, and S. Lajoie. *Toward a cognitive theory for the measurement of achievement*, pages 41–85. R. Ronning and J.A. Glover and J.C. Conoley and J.C. Witt (eds.), Hillsdale, NJ: Lawrence Erlbaum Associates, 1987.

A. Gliozzo, B. Magnini, and C. Strapparava. Unsupervised Domain Relevance Estimation for Word Sense Disambiguation. *Proceedings of the Empirical Methods in Natural Language Processing Conference, Barcelona, Spain*, 2004.

A. Gliozzo, C. Giuliano, and C. Strapparava. Domain Kernels for Word Sense Disambiguation. *Ann Arbor*, 100, 2005.

E. Gouli, K. Papanikolaou, and M. Grigoriadou. Personalizing Assessment in Adaptive Educational Hypermedia Systems. *Adaptive Hypermedia and Adaptive Web-based Systems, Second International Conference, LNCS*, 2346:153–163, 2002.

E. Gouli, A. Gogoulou, K. Papanikolaou, and M. Grigoriadou. COMPASS: An adaptive web-based concept map assessment tool. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain, 2004.

A.C. Graesser, P. Chipman, B.C. Haynes, and A. Olney. Autotutor: an Intelligent Tutoring System with mixed-initiative dialogue. *IEEE Transactions on Education*, 48(4):612–618, 2005.

C. Guetl, H. Dreher, and R. Williams. E-tester: A computer-based tool for auto-generated question and answer assessment. *E-Learn*, 2005.

S. Gutiérrez, A. Pardo, and C. Delgado. An adaptive tutoring system based on hierarchical graphs. In *Adaptive Hypermedia and Adaptive Web-Based Systems. Proceedings of the AH'2004 conference. LNCS 3137*, pages 401–404, Heidelberg, 2004. Springer-Verlag.

E. Guzmán and R. Conejo. An adaptive assessment tool integrable into internet-based learning systems. In *Educational Technology: International Conference on TIC's in Education*, volume 1, pages 139–143, 2002.

D. Haley, P. Thomas, B. Nuseibeh, J. Taylor, and P. Lefrere. E-assessment using Latent Semantic Analysis. In *LeGE-WG 3*, 2003.

R.K. Hambleton, H. Swaminathan, and H.J. Rogers. *Fundamentals of Item Response Theory*. Sage Publications, 1991.

N. Hara. Student distress in a web-based distance education course. *Information Communication & Society*, 3(4):557–579, 2000.

D. Hartley and A. Mitrovic. The effectiveness of open student modelling on learning, 2001.

D. Hartley and A. Mitrovic. Supporting learning by opening the student model. *Intelligent Tutoring Systems, 6th International Conference, ITS 2002, Biarritz, France and San Sebastian, Spain, June 2-7, 2002, Proceedings*, 2363:453–462, 2002.

M. Hearst. The debate on automated essay grading. *IEEE Intelligent Systems*, 5(15):22–37, 2000.

D. Helic, H. Maurer, and N. Scherbakov. Web based training: What do we expect from the system. *Proceedings of ICCE*, pages 1689–1694, 2000.

N. Henze, W. Nejdl, and M. Wolpers. Modeling Constructivist Teaching Functionality and Structure in the KBS Hyperbook System. In *Proceedings of Computer Supported Collaborative Learning Conference*, 1999.

T. Hofmann. Unsupervised learning by Probabilistic Latent Semantic Analysis. *Machine Learning*, 42:177–196, 2001.

J.H. Holland. *Adaptation in natural and artificial systems*. MIT Press Cambridge, MA, USA, 1992.

P.B. Horton, A.A. McConney, M. Gallo, A.L. Woods, G.J. Senn, and D. Hamelin. An investigation of the effectiveness of concept mapping as an instructional tool. *Science Education*, 77(1):95–111, 1993.

L. Hsu, R. Edd, S. Hsieh, and R. Msn. Concept maps as an assessment tool in a nursing course. *Journal of Professional Nursing*, 21(3):141–149, 2005.

S. Huang. A content-balanced adaptive testing algorithm for computer-based training systems. In C. Frasson, G. Gauthier, and A. Lesgold, editors, *Intelligent Tutoring Systems, Third International Conference*, pages 306–314. Springer, 1996.

G.J. Hwang. A conceptual map model for developing Intelligent Tutoring Systems. *Computers and Education*, 40:217–235, 2003.

B. Inherlder and J. Piaget. *The early growth of logic in the child*. New York: Basic Books, 1964.

T. Ishioka and M. Kameda. Automated Japanese Essay Scoring System: Jess. *Proceedings of the 15th International Workshop on Database and Expert Systems Applications*, pages 4–8, 2004.

T. Ishioka and M. Kameda. Automated Japanese Essay Scoring System based on Articles Written by Experts. In *Proceedings of the ACL*, 2006.

A. Jameson. User-adaptive systems: An integrative overview, 1999.

P.N. Johnson-Laird. Mental models: Towards a cognitive science of language. *Inference, and Consciousness, Cambridge*, 1983.

D.H. Jonassen. *Computers as mindtools for schools: Engaging critical thinking (2nd ed.)*. New Jersey: Prentice Hall, 2000.

B.F. Jones, A.S. Palinscar, D.S. Ogle, and E.G. Carr. Strategic teaching and learning: Cognitive instruction in the content areas. Elmhurst, IL: North Central Regional Laboratory and the Association for Supervision and Curriculum Development, 1987.

S.S. Jones and L.B. Smith. The place of perception in children's concepts. *Cognitive Development*, 8(2):113–39, 1993.

J.S. Justeson and S.L. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 3(2):259–289, 1996.

T. Kakkonen, N. Myller, J. Timonen, and E. Sutinen. Automatic Essay Grading with Probabilistic Latent Semantic Analysis. In *Proceedings of the 2nd Workshop on Building Educational Applications Using NLP, Association for Computational Linguistics*, pages 29–36, 2005.

T. Kakkonen, N. Myller, and E. Sutinen. Applying Latent Dirichlet Allocation to automatic essay grading. *Advances In Natural Language Processing, Proceedings*, 4139:110–120, 2006.

M. Kang and H.P. Byun. A conceptual framework for a web-based knowledge construction support system. *Educational Technology*, 41, 2001.

A. Kavcic. Enhancing educational hypermedia: Personalization through fuzzy logic. In *1st COST 276 Workshop*, 2001.

E. Kosba, V. Dimitrova, and R. Boyle. Using student and group models to support teachers in web-based distance education. *Proceedings of the User Modeling International Conference*, 3538:124–133, 2005.

H.C. Kraemer. *Encyclopedia of Statistical Sciences*, chapter Kappa coefficient. New York: John Wiley & Co, 1982.

M. Kravcik and M. Specht. Flexible Navigation Support in the WINDS Learning Environment for Architecture and Design. *Proceedings of the AH 2004 Conference*, 2004.

R. Kremer. Concept mapping: Informal to formal. In *Proceedings of the International Conference on Conceptual Structures (ICCS)*, Maryland, U.S.A., 1994.

R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 191–203, 1993.

Z. Kunda. *Social cognition*. MIT press, Cambridge, Massachusetts, London, England, 1999.

S. Labidi and N. Sergio. Student modeling and semi-automatic domain ontology construction for shiecc. In *Proceedings of the 30th ASEE/IEEE Frontiers in Education Conference*, 2000.

T.K. Landauer and S.T. Dumais. A solution to PLATO's problem: The Latent Semantic Analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240, 1997.

T.K. Landauer, D. Laham, B. Rehder, and M.E. Schreiner. How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and humans. In M. G. Shafto and P. Langley, editors, *Proceedings of the 19th annual meeting of the Cognitive Science Society*, pages 412–417, 1997.

T.K. Landauer, P.W. Foltz, and D. Laham. An introduction to Latent Semantic Analysis. *Discourse Processes*, 25(2):259–284, 1998.

T.K. Landauer, D. Laham, and P.W. Foltz. The Intelligent Essay Assesor: putting knowledge to the test. In *Proceedings of the Association of Test Publishers Computer-Based Testing: Emerging Technologies and Opportunities for Diverse Applications conference*, 2001.

L. Larkey, S. Croft, and W. Bruce. *A Text Categorization Approach to Automated Essay Grading*, pages 55–70. Lawrence Erlbaum, 2003.

L. S. Larkey. Automatic Essay Grading Using Text Categorization Techniques. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 90–95, 1998.

C. Leacock. Scoring free-responses automatically: A case study of a large-scale assessment. English version of Leacock, C. (2004). Automatisch beoordelen van antwoorden op open vragen; een taalkundige benadering. Examens, 2004.

D. Leake, A. Maguitman, T. Reichherzer, A.J. Caas, M. Carvalho, M. Arguedas, and T. Eskridge. Googling from a concept map: towards automatic concept-map-based query formation. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain, 2004.

D. Levine. Users guide to the pgapack parallel genetic algorithm library. *Argonne National Laboratory*, 95(18):1–77, 1996.

K. Lewin. *Will and needs: A source book of Gestalt psychology.* London: Routledge and Kegan Paul Ltd., 1969.

D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–50, 1992.

M. Lilley and T. Barker. An evaluation of a Computer Adaptive Test in a UK University Context. In *Proceedings of the 7th Computer Assisted Assessment Conference*, 2003.

M. Lilley, T. Barker, and C. Britton. Automated feedback for a Computer-Adaptive Test: a case study. In *Proceedings of the 9th International Computer Assisted Assessment (CAA) conference*, 2005.

C. Lin and E. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the Human Technology Conference 2003 (HLT-NAACL-2003)*, 2003.

J.J. Little. Computerized evaluation of essays. Term Paper, Expert Systems, 2001.

R. Lutticke. Problem solving with adaptive feedback. In *Adaptive Hypermedia and Adaptive Web-Based Systems. Proceedings of the AH'2004 conference. LNCS 3137*, pages 417–420, Heidelberg, 2004. Springer-Verlag.

R. Lutticke. Graphic and NLP Based Assessment of Knowledge about Semantic Networks. In *Proceedings of the Artifical Intelligence in Education (AIED) conference*, 2005.

G.D. Magoulas, K.A. Papanikolaou, and M. Grigoriadou. Neuro-fuzzy synergism for planning the content in a web-based course. *Informatica*, 25(1):39–48, 2001.

K. Malatesta, P. Wiemer-Hastings, and J. Robertson. Beyond the short answer question with research methods tutor. In *Proceedings of the Intelligent Tutoring Systems Conference*, 2002.

D. Marcu. The theory and practice of discourse parsing and summarization. The MIT Press, 2000.

O. Mason and I. Grove-Stephenson. Automated free text marking with paperless school. In *Proceedings of the 6th International Computer Assisted Assessment Conference*, 2002.

A. Maynard and S. Ananiadou. Identifying terms by their family and friends. In *Proceedings of COLING*, pages 530–536, Saarbrucken, Germany, 2000.

R. Mazza and V. Dimitrova. Generation of graphical representations of student tracking data in course management systems. In *Proceedings of the ninth International Conference on Information Visualisation*, pages 253–258, 2005.

R. Mazza and C. Milani. Gismo: a graphical interactive student monitoring tool for course management systems. *In Proceedings of the Technology Enhanced Learning International Conference*, pages 18–19, 2004.

A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, 1998.

P. Mcgrath. Assessing Students: Computer Simulation vs MCQs. In *Proceedings of the 7th Computer Assisted Assessment Conference*, 2003.

D.L. Medin, R.L. Goldstone, and D. Gentner. Respects for similarity. *Psychological Review*, 100:254 – 278, 1993.

D.L. Medin, E.B. Lynch, and K.O. Solomon. Are there kinds of concepts? *Annual Review Psychology*, 51:121–147, 2000.

A. Mikhailov. Indextron. *Intelligent Engineering Systems Through Artificial Neural Networks*, 8:57, 1998.

E. Millan-Valldeperas. *Bayesian System for Student Modelling.* PhD thesis, University of Malaga, 2000.

Y. Ming, A. Mikhailov, and T.L. Kuan. Intelligent Essay Marking System. *Learners Together, NgeeANN Polytechnic, Singapore*, 2000.

J. Mintzes, J. Wandersee, and J. Novak. *Assessing Science Understanding.* Academic Press, San Diego, 2000.

T. Mitchell. *Machine Learning.* WCB/McGraw-Hill, 1997.

T. Mitchell, T. Russell, P. Broomhead, and N. Aldridge. Towards robust computerised marking of free-text responses. In *Proceedings of the 6th Computer Assisted Assessment Conference*, 2002.

T. Mitchell, N. Aldridge, W. Williamson, and P. Broomhead. Computer based testing of medial knowledge. In *Proceedings of the 7th Computer Assisted Assessment Conference*, 2003.

A. Mitrovic. Cosc420 lecture notes: Cognitive modeling and intelligent tutoring systems, 2001.

A. Mitrovic and B. Martin. Evaluating adaptive problem selection. In *Adaptive Hypermedia and Adaptive Web-Based Systems. Proceedings of the AH'2004 conference. LNCS 3137*, pages 185–194, Heidelberg, 2004. Springer-Verlag.

MUC7. *Proceedings of the 7th Message Understanding Conference (MUC-7)*. Morgan Kaufman, 1998.

M. Muehlenbrock, S. Winterstein, E. Andres, and A. Meier. Learner modeling in iClass. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia, and Telecommunications EdMedia*, Montreal, Canada, 2005.

G. L. Murphy. *The big book of concepts*. A Bradford Book, MIT Press, 2002.

J. D. Novak. Concept maps and Vee diagrams: two metacognitive tools for science and mathematics education. *Instructional Science*, 19:29–52, 1990.

J.D. Novak. *A Theory of Education*. Cornell University Press, Ithaca, New York, 1977.

J.D. Novak and A. J. Canas. The theory underlying concept maps and how to construct them. Technical report, Florida Institute for Human and Machine Cognition, 2006.

J.D. Novak and D.B. Gowin. *Learning How to Learn*. Cambridge University Press, Cambridge, U.K., 1984.

J.D. Novak, D.B. Gowin, and C.T. Johansen. The Use of Concept Mapping and Knowledge Vee Mapping with Junior High School Science Students. *Science Education*, 67(5):625–645, 1983.

J. Olea. *Evaluación del aprendizaje en contextos universitarios*. Psychology Department, Universidad Autónoma de Madrid, 2006.

E.B. Page. The imminence of grading essays by computer. *Phi Delta Kappan*, 47(1):238–243, 1966.

E.B. Page. Computer grading of student prose, using modern concepts and software. *Journal of Experimental Education*, 2(62):127–142, 1994.

K. Palmer and P. Richardson. On-line assessment and free-response input - a pedagogic and technical model for squaring the circle. In *Proceedings of the 7th Computer Assisted Assessment Conference*, 2003.

P. Pantel and L. Dekang. A statistical corpus-based term extractor. In *Proceedings of the 14th Biennial Conference of the Canadian Society on Computational Studies of Intelligence*, pages 36–46, London, UK, 2001. Springer-Verlag.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. Research report, IBM, 2001.

H. Parsons, D. Schofield, and S. Woodget. Piloting summative web assessment in secondary education. In *Proceedings of the 7th Computer Assisted Assessment Conference*, 2003.

D. Pérez. Automatic evaluation of users' short essays by using statistical and shallow Natural Language Processing techniques. Master's thesis, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 2004.

D. Pérez and E. Alfonseca. Application of the BLEU algorithm for recognising textual entailments. In *Proceedings of the Recognising Textual Entailment Pascal Challenge*, 2005e.

D. Pérez and E. Alfonseca. Using BLEU-like Algorithms for the Automatic Recognition of Entailment. *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification and Recognising Textual Entailment, LNCS*, 3944, 2006e.

D. Pérez, E. Alfonseca, and P. Rodríguez. Adapting the Automatic Assessment of free-text Answers to the Students' Profiles. In *Proceedings of the International Computer Assisted Assessment conference*, Loughborough, U.K., 2005a.

D. Pérez, A. Gliozzo, C. Strapparava, E. Alfonseca, P. Rodríguez, and B. Magnini. Automatic Assessment of Students' free-text Answers underpinned by the Combination of a BLEU-inspired algorithm and Latent Semantic Analysis. In *Florida Artificial Intelligence Research Society conference, FLAIRS-2005*. American Association for Artificial Intelligence (AAAI) Press, 2005b.

D. Pérez, A. Gliozzo, E. Alfonseca, C. Strapparava, B. Magnini, and P. Rodríguez. About the effects of combining Latent Semantic Analysis with other Natural Language Processing techniques to assess open-ended questions. *Revista Signos*, 38(59), 2005c.

D. Pérez, O. Postolache, E. Alfonseca, D. Cristea, and P. Rodríguez. About the effects of using anaphora resolution in assessing free-text student answers. In *Proceedings of the international conference of Recent Advances in Natural Language Processing (RANLP)*, 2005d.

D. Pérez, E. Alfonseca, and P. Rodríguez. Can computers assess open-ended questions? *Revista Novática*, 183, 2006f. Asociación de técnicos en informática (ATI).

D. Pérez-Marín, E. Alfonseca, and P. Rodríguez. A free-text scoring system that generates conceptual models of the students knowledge with the aid of clarifying questions. In *Proceedings of the International WorkShop Semantic Web technologies for E-Learning (SWEL)*, 2006b.

D. Pérez-Marín, E. Alfonseca, P. Rodríguez, and I. Pascual-Nieto. Willow: Automatic and adaptive assessment of students free-text answers. In *Proceedings of the 22nd International Conference of the Spanish Society for the Natural Language Processing (SEPLN)*, 2006c.

D. Pérez-Marín, E. Alfonseca, P. Rodríguez, and I. Pascual-Nieto. Automatic generation of students' conceptual models from answers in plain text. In *To appear in the User Modeling 2007 International Conference, LNAI, Springer-Verlag*, 2007a.

D. Pérez-Marín, I. Pascual-Nieto, E. Alfonseca, and P. Rodríguez. Automatically Generated Inspectable Learning Models for Students. In *To appear in the proceedings of the international conference Artificial Intelligence in Education (AIED)*, 2007b.

D. Pérez-Marín, I. Pascual-Nieto, E. Alfonseca, E. Anguiano, and P. Rodríguez. A study on the impact of the use of an automatic and adaptive free-text assessment system during a university course. In *To appear in the Blended Learning Pearson book*, 2007c.

E. Plotnick. Concept mapping: A graphical system for understanding the relationship between concepts. In *ERIC Clearinghouse on Information and Technology, ED407938*, New York, U.S.A., 1997.

M.F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

S. Puntambekar, A. Stylianou, and R. Hubscher. Improving navigation and learning in hypertext environments with navigable concept maps. *Human-Computer Interaction*, 18(4):395–428, 2003.

J.R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers: San Mateo, CA., 1993.

L.C. Ragan. Good Teaching Is Good Teaching. An Emerging Set of Guiding Principles and Practices for the Design and Development of Distance Education. *Cause/Effect*, 22(1):20–24, 1999.

F.E.L. Rocha and E.L. Favero. CMTool: A Supporting Tool for Conceptual Map Analysis. In *Proceedings of World Congress on Engineering and Technology Education*, Santos, Brazil, 2004.

E. Rosch. *Principles of categorization*, page 2748. Hillsdale: Lawrence Erlbaum, 1978.

C.P. Rosé, A. Roque, D. Bhembe, and K. VanLehn. A hybrid text classification approach for analysis of student essays. In *HLT-NAACL Workshop on Building Educational Applications Using Natural Language Processing*, pages 68–75, 2003.

B. Ross and H. Munby. Concept mapping and misconceptions: a study of highschool students understanding of acids and bases. *International Journal of Science Education*, 13:11–24, 1991.

C. Rovira. El editor de mapas conceptuales digidocmap y la norma topic maps. [on line] http://www.hipertext.net, 2005.

L. Rudner and P. Gagne. An overview of three approaches to scoring written essays by computer. Educational Resources Informaton Center (ERIC) digest, 12 2001.

L.M. Rudner and T. Liang. Automated Essay Scoring Using Bayes' Theorem. In *Proceedings of the annual meeting of the National Council on Measurement in Education*, 2002.

L.M. Rudner, V. Garcia, and C. Welch. An Evaluation of the IntelliMetric Essay Scoring System. *Journal of Technology, Learning, and Assessment*, 4(4), 2006.

U. Rueda, M. Larrañaga, A. Arruarte, and J.A. Elorriaga. Modelado de grupos en actividades de aprendizaje basado en mapas conceptuales. *Revista Iberoamericana de Inteligencia Artificial*, 8(24):131–140, 2004.

M. Ruiz-Casado, E. Alfonseca, and P. Castells. Automatic assignment of Wikipedia encyclopedic entries to Wordnet synsets. *Proceedings of the Atlantic Web Intelligence Conference (AWIC)*, 3528:380–386, 2005.

M.A. Ruiz-Primo. Examining concept maps as an assessment tool. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain, June 2004.

M.A. Ruiz-Primo and R.J. Shavelson. Problems and issues in the use of concept maps in science assessment. *Journal of Research in Science Teaching*, 33(6):569–600, 1996.

F. Safayeni, N. Derbentseva, and A.J. Cañas. Concept maps: A theoretical note on concepts and the need for cyclic concept maps. *Journal of Research in Science Teaching*, 2003.

G. Salton, A. Wong, and C.S. Yang. A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 11(18):613–620, 1975.

F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

J.J. Sellers. An empirical evaluation of a fault-tolerant approach to computer-assisted teaching of binary relations. Master's thesis, North Carolina State University., 1998.

K. Seta, M. Ikeda, O. Kakusho, and R. Mizoguchi. Capturing a conceptual model for end-user programming: Task ontology as a static user model. In A. Jameson, C. Paris, and C. Tasso, editors, *Proceedings of the Sixth International Conference on User Modeling (UM)*, New York, U.S.A., 1997. Springer.

M.D. Shermis, C.M. Koch, E.B. Page, T.Z. Keith, and S. Harrington. Trait rating for automated essay scoring. *Educational and Psychological measures*, 62:5–18, 2002.

V.J. Shute and L.A. Torreano. Formative evaluation of an automated knowledge elicitation and organization tool. *Authoring Tools for Advanced Technology Learning Environments: Toward Cost-Effective Adaptive, Interactive, and Intelligent Educational Software*, 2002.

I.E. Sigel, editor. *Development of mental representations: Theories and Applications*. Lawrence Erlbaum Associates, New Jersey, U.S.A., 1999.

J.E. Sims-Knight, R.L. Upchurch, N. Pendergrass, T. Meressi, P. Fortier, P. Tchimev, R. VonderHeide, and M. Page. Using concept maps to assess design process knowledge. In *Proceedings of the 34th ASEE/IEEE Frontiers in Education (FIE) conference*, Georgia, U.S.A, October 2004.

D.K. Sleator and D. Temperley. *Parsing English with a link grammar*. School of Computer Science, Carnegie Mellon University, 1991.

K. Smith-Gratto. Distance education best practices and problems North Carolina A & T state university. Technical report, Report to the Distance Education Evaluation Task Force, North Carolina A & T State University, 1999.

F. Sormo. Case-based student modeling using concept maps. *Case-Based Reasoning Research And Development, Proceedings*, 3620:492–506, 2005.

S. Sosnovsky. Adaptive navigation for self-assessment quizzes. In *Adaptive Hypermedia and Adaptive Web-Based Systems. Proceedings of the AH'2004 conference. LNCS 3137*, pages 365–371, Heidelberg, 2004. Springer-Verlag.

J.F. Sowa. Conceptual Structures: Information Processing in Mind and Machine. *Adison-Wesley*, 1984.

L. Streeter, J. Pstoka, D. Laham, and D. MacCuish. The credible grading machine: Automated Essay Scoring in the DoD. In *Proceedings of Interservice/Industry, Simulation and Education Conference (I/ITSEC)*, 2003.

J.Z. Sukkarieh, S.G. Pulman, and N. Raikes. Auto-marking: using computational linguistics to score short, free text responses. In *Proceedings of the 29th IAEA Conference, theme: Societies' Goals and Assessment*, 2003.

J.R. Taylor. *Linguistic categorization*. Oxford: Clarendon Press, 1995.

P. Thagard. *Conceptual revolutions*. NJ: Princeton University Press, 1992.

A. Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

L.H. Ungar and D.P. Foster. Clustering methods for collaborative filtering. In Wisc. Madison, editor, *AAAI-98 Workshop on Recommender Systems*, 1998.

S. Valenti, F. Neri, and A. Cucchiarelli. An overview of current research on automated essay grading. *Journal of Information Technology Education*, 2:319–330, 2003.

C. J. van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

K. VanLehn, P. Jordan, and C.P. Rosé. The architecture of Why2-Atlas: a coach for qualitative physics essay writing. In *Proceedings of the Intelligent Tutoring Systems Conference*, 2002.

Vantage. A study of expert scoring and IntelliMetric scoring accuracy for dimensional scoring of grade 11 student writing responses. Technical Report RB-397, Vantage Learning Tech., 2000.

Vantage. A preliminary study of the efficacy of IntelliMetric for use in scoring hebrew assessments. Technical Report RB-561, Vantage Learning Technologies, 2001.

P. Vossen. *EuroWordNet - A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, 1998.

A. Voutilainen. Nptool, a detector of English noun phrases. In *Proceedings of the Workshop on Very Large Corpora*, pages 48–57, 1993.

H. Wainer. *Computerized Adaptive Testing: A Primer*. Lawrence Erlbaum Assoc Inc, 2000.

G.I. Webb, M.J. Pazzani, and D. Billsus. Machine Learning for User Modeling. *User Modeling and User-Adapted Instruction*, 11, 2001.

E. Wenger. *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1987.

C.K. West, J.A. Farmer, and P.M. Wolff. *Instructional design: Implication from cognitive science*. Prentice Hall, New Jersey, U.S.A., 1991.

D. Whittingdon and H. Hunt. Approaches to the computerised assessment of free-text responses. In *Proceedings of the 3rd International Computer Assisted Assessment Conference*, 1999.

P. Wiemer-Hastings, A.C. Graesser, D. Harter, and the Tutoring Research Group. The foundations and architecture of AutoTutor. In *Proceedings of the 4th International Conference on Intelligent Tutoring Systems*, pages 334–343. Springer-Verlag, 1998.

P. Wiemer-Hastings, D. Allbritton, and E. Arnott. RMT: A Dialog-Based Research Methods Tutor with or without a Head. In *Proceedings of the ITS2004 Seventh International Conference*, Berlin, 2004. Springer.

P. Wiemer-Hastings, E. Arnott, and D. Allbritton. Initial results and mixed directions for research methods tutor. In *Proceedings of the Artifical Intelligence in Education (AIED) conference*, 2005.

R. Williams. Automated essay grading: an evaluation of four conceptual models. In Herrmann and M. M. Kulski, editors, *Proceedings of the 10th Annual Teaching Learning Forum*, 2001.

R. Williams and H. Dreher. Automatically Grading Essays with Markit. In *Proceedings of Informing Science 2004 Conference*, Rockhampton, Queensland, Australia, 2004.

L. Wittgenstein. *Philosophical investigations (3rd ed.)*. Prentice Hall, Englewood Cliffs, NJ., 1958.

S.K.M. Wong, W. Ziarko, V. V. Raghavan, and P. C.N. Wong. On modeling of Information Retrieval concepts in vector spaces. *ACM Trans. Database Syst.*, 12(2):299–321, 1987. ISSN 0362-5915. doi: http://doi.acm.org/10.1145/22952.22957.

Idea Works. Sagrader, 2006. http://www.ideaworks.com/sagrader/index.html.

L.A. Zadeh. Fuzzy sets and systems. *Information and Control*, 8(3):338–353, 1965.

J.D. Zapata-Rivera. Interacting with inspectable bayesian student models. *International Journal of Artificial Intelligence in Education*, 14(2):127–163, 2004.

J.D. Zapata-Rivera and J.E. Greer. Externalising learner modelling representations. *Proceedings of Workshop on External Representations of AIED: Multiple Forms and Multiple Roles*, pages 71–76, 2001.

D. M. Zimmaro and J. M. Cawley. Concept map module, 1998.

Web References

http1 My web page, http://orestes.ii.uam.es:8080/dperez/spanish/research.html

http2 Willow, http://orestes.ii.uam.es:8080/ateneaAdaptativa/jsp/loginAtenea.jsp

http3 Willed, http://orestes.ii.uam.es:8080/ateneaAdaptativa/jsp/loginEditor.jsp

http4 Willoc, http://orestes.ii.uam.es:8080/ateneaAdaptativa/jsp/loginConfigurador.jsp

http5 COMOV, http://orestes.ii.uam.es:8080/ateneaAdaptativa/jsp/loginGeneraModeloConceptual.jsp

http6 Moodle, http://moodle.org/

http7 Automark, http://www.intelligentassessment.com

http8 ExamOnline, http://www.examonline.co.uk

http9 Auto-marking, http://www.ucles.org.uk/

http10 C-rater and E-rater, http://www.ets.org

http11 EGAL, https://sourceforge.net/projects/egal/

http12 WordNet, http://wordnet.princeton.edu/

http13 IEA, http://www.knowledge-technologies.com

http14 IntelliMetric, http://www.vantage.com

http15 Jess, http://coca.rd.dnc.ac.jp/jess/

http16 MarkIT, http://www.essaygrading.com/index.jspx

http17 MRW, http://pi7.fernuni-hagen.de/

http18 PEG, http://134.68.49.185/pegdemo/ref.asp

http19 PS-ME, http://www.paperless-school.com

http20 SAGrader, http://www.ideaworks.com

http21 SEAR, http://www.comp.rgu.ac.uk/staff/jrc/fSEAR.htm

http22 Wraetlic NLP tools, http://www.ii.uam.es/~ealfon/esp/research/wraetlic.html

http23 Babelfish, http://babelfish.altavista.com/

http24 BLEU, http://www.nist.gov/speech/tests/mt/resources/scoring.htm

http25 PGAPack, http://www-fp.mcs.anl.gov/

http26 Google Glossary, http://labs.google.com/