Automatic Assessment of Students' Free-text Answers Underpinned by the Combination of a BLEU-inspired Algorithm and Latent Semantic Analysis

Diana Pérez[1], **Alfio Gliozzo**[2], **Carlo Strapparava**[2],
Enrique Alfonseca[1], **Pilar Rodríguez**[1] and **Bernardo Magnini**[2]

[1]Computer Science Department, Universidad Autonoma de Madrid (UAM), 28043 Madrid, Spain
{diana.perez,enrique.alfonseca,pilar.rodriguez}@uam.es

[2]Istituto per la Ricerca Scientifica e Tecnologica (IRST), 38050 Trento, Italy
{gliozzo,strappa,magnini}@itc.it

Abstract

In previous work we have proved that the BLEU algorithm (Papineni *et al.* 2001), originally devised for evaluating Machine Translation systems, can be applied to assessing short essays written by students. In this paper we present a comparative evaluation between this BLEU-inspired algorithm and a system based on Latent Semantic Analysis. In addition we propose an effective combination schema for them. Despite the simplicity of these shallow NLP methods, they achieve state-of-the-art correlations to the teachers' scores while keeping the language-independence and without requiring any domain specific knowledge.

Introduction

Assessing students' answers is a very time-consuming activity that makes teachers cut down the time they can devote to other duties. In some cases, they may even have to reduce the number of assignments given to their students due to lack of time. Many researchers believe that computers can be used to help the teachers in their assessment task. This is the basis of the field known as Computer-Assisted Assessment (CAA) of free-text answers.

CAA of free-text answers is a long-standing problem that has attracted interest from the research community since the sixties (Page 1966) and has not been fully solved yet. On the other hand, the success of e-learning and the advances in other areas such as Information Extraction (IE) and Natural Language Processing (NLP) have made CAA of free-text answers a flourishing research line in the last few years. A computer can examine and analyze essays in much more detail than a human teacher, as is totally free of any kind of judgements, myths, false beliefs and value biases (Streeter *et al.* 2003). In the literature, several techniques have been used to tackle this problem with increasingly better results. They can be grouped into five main categories: statistical, NLP, IE, clustering, and integrated-approaches (Valenti, Neri, & Cucchiarelli 2003).

In previous work we presented Atenea (Alfonseca & Pérez 2004), a system for scoring automatically open ended questions. According to the already introduced classification, it is an integrated approach, since it relies on the combination of a BLEU (Papineni *et al.* 2001) inspired algorithm, called Evaluating Responses with BLEU (ERB) (Pérez, Alfonseca, & Rodríguez 2004a), with a set of NLP techniques such as stemming, closed-class word removal, Word Sense Disambiguation and synonyms treatment procedures. In particular, the ERB algorithm performs a syntactic analysis of the students' answers.

However, we believe that, in order to fully assess the answers, both a syntactic and a semantic analysis is required. Thus we decided to include a semantic module in Atenea, based on Latent Semantic Analysis (LSA). LSA has been successfully applied to evaluating free-text answers, as reported in (Foltz, Laham, & Landauer 1999; Dessus, Lemaire, & Vernier 2000). Then we exploited Atenea's infrastructure to combine ERB with LSA, that constitutes the original contribution of this paper. Experimental results confirm the hypothesis of the natural complementarity of syntax and semantics, by reporting a significant improvement in the accuracy of the combined system.

It is usually very difficult to perform a comparison of free-text CAA systems due to the lack of common corpora and standard metrics. Nevertheless, because in our case we can apply the same experimental settings for the LSA and the ERB techniques, we have the possibility of doing for the first time a comparative analysis of both ERB and LSA in a common set of students' and teacher's answers. It is useful to perform this comparison in order to study how different statistical approaches can fulfill the same goal. For evaluation purposes, the Pearson correlation coefficient between the humans' scores and the system's scores is calculated.

It is also interesting to study how far we can go in the task of assessing students' answers by only exploiting "shallow" NLP techniques, such as ERB and LSA. Shallow NLP techniques can be easily implemented for any language. Typically, they do not require "ad-hoc" lexical resources and domain specific knowledge. The only resource required by both ERB and LSA is a corpus of students' answers. LSA also exploits a large collection of domain specific texts to induce lexical knowledge in a totally unsupervised way. In the Web era, collections of non-annotated texts are easily available.

The paper is organized as follows: in Section *Atenea* we give a general overview of the internal architecture of Atenea to focus in the description of both the ERB and the LSA modules. In the same section we describe the system combination strategy we adopted to combine ERB and LSA. In Section *Experimental settings* we describe the corpora used for training and evaluating our systems. In section *Evaluation* we compare the performances of all the "basic" modules and their combination. Finally, in the last section, we draw some conclusions and discuss some future developments.

Atenea

Atenea (Pérez, Alfonseca, & Rodríguez 2004a; Pérez, Alfonseca, & Rodríguez 2004b; Alfonseca & Pérez 2004) is a CAA system for automatically scoring students' short answers. It was developed as a web-based application so that it can be accessed through any web browser connected or not to the Internet.

In order to assign the scores, Atenea has access to a database of questions associated to a set of *references* (i.e. free-text answers written by teachers). As an option, the best students' answers can be included in the reference set (Pérez, Alfonseca, & Rodríguez 2004b). Each time a student logs into the system, Atenea asks him or her a question chosen from the database in a random way or depending on the student's profile (Alfonseca *et al.* 2004) and compares the answer typed by the student with the associated references.

The internal architecture of Atenea is composed of the ERB module and of several shallow NLP modules. First of all, both the student's answer and the reference answers are tokenized. Secondly, the "basic" modules (i.e. Blue, ERB, LSA) are invoked to independently assign a score to the answer. Finally, their outputs are combined to compose the final score provided by the system to the student's answer.

The framework provided by Atenea allows us to independently evaluate both the ERB and the LSA algorithms and to combine them, just by setting up the Atenea's configuration file. In the following subsections we will describe the two basic algorithms (ERB and LSA) we used for our experiments, then we will introduce a framework for combining them.

ERB

The ERB algorithm compares the student's answer and the references using a modified version of the n-gram co-occurrence scoring algorithm called BLEU (Papineni *et al.* 2001). The core idea of these algorithms is that the more similar a student's answer (the *candidate* text) is to the teachers' answers (the *references*), the better it is, and, consequently, it will have a higher score.

BLEU uses a Modified Unified Precision (MUP) metric that clips the frequency of the n-gram according to the number of times it appears in the candidate and in the references. MUP must be calculated for each value of n, which usually ranges from 1 to 4. For longer n-grams from the candidate text, it will be unlikely to find them in the references. Next, a weighted sum of the logarithms of MUPs is performed. In the last step, a penalization is applied to very short answers, which might be incomplete, by multiplying the previous value by a Brevity Penalty (BP) factor.

We have modified the original algorithm so that it takes into account not only the precision (the original BLEU score) but also the recall that is calculated by studying the percentage of the references that is covered by the student's answer (Pérez, Alfonseca, & Rodríguez 2004b), using a Modified Brevity Penalty (MBP) factor. We have called this BLEU-inspired algorithm *Evaluating Responses with* BLEU (ERB). Equation 1 shows the final formula for calculating the score of an answer a. n represents the length of the n-grams, and, N is the highest value than n can take.

$$ERB_{score}(a) = MBP(a) \times e^{\sum_{n=0}^{N} \frac{log(MUP(n))}{N}} \quad (1)$$

LSA

LSA (Deerwester *et al.* 1990; Foltz, Kintsch, & Landauer 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 of the terms-by-documents matrix D extracted from a large scale corpus. In other words, term co-occurrences in the corpus are captured by means of a dimensionality reduction operated on the matrix D. The vectors in the original space are mapped into a lower dimensional space, in which the sparseness problem disappears, and similarity estimation is more accurate. The resulting LSA vectors can be exploited to estimate both term and document similarity.

Regarding document similarity, we used a variation of the *pseudo-document* methodology described in (Berry 1992), 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).

It has been claimed (Deerwester *et al.* 1990) that, 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.

The LSA algorithm we have 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

SET	NC	MC	NR	MR	Type	Desc
1	38	67	4	130	Def.	OS
2	79	51	3	42	Def.	OS
3	96	44	4	30	Def.	OS
4	11	81	4	64	Def.	OOP
5	143	48	7	27	A/D	OS
6	295	56	8	55	A/D	OS
7	117	127	5	71	Y/N	OS
8	117	166	3	186	A/D	OS
9	14	118	3	108	Y/N	OS
10	14	116	3	105	Def.	OS

Table 1: Evaluation datasets. Columns indicate: set number; number of candidate texts (NC), their mean length (MC), number of reference texts (NR), their mean length (MR), question type (Def = definitions; A/D = advantages/disadvantages; Y/N = justified Yes/No), and a short description (OS = Operating System exam question; OOP = Object-Oriented Programming exam question).

SET	BLEU	ERB	SA-LSA	CS-LSA
1	0.59	0.61	**0.71**	0.49
2	0.29	**0.54**	0.39	0.20
3	**0.22**	0.20	0.17	-0.01
4	**0.73**	0.29	-0.22	0.52
5	0.35	0.61	**0.69**	0.50
6	0.04	0.19	**0.27**	0.24
7	0.23	**0.33**	0.07	0.29
8	0.27	**0.39**	0.34	**0.39**
9	0.09	0.75	0.66	**0.78**
10	0.26	0.78	**0.91**	0.87
Mean	0.31	**0.47**	0.40	0.43

Table 2: Evaluation of BLUE, ERB, SA-LSA and CS-LSA. The first column indicates the question number, the following ones report the correlation to humans' scores achieved by BLEU, ERB, SA-LSA and CS-LSA. The last row reports the mean correlations.

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 2.

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

The combination of ERB and 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. The independence of the systems outputs is a fundamental prerequisite to combine them, so it will be checked in the evaluation. If it is satisfied, it is expected that the performances of independent classifiers will be increased by adopting a system combination schema (Florian *et al.* 2002). The combination schema we adopted for our experiments is the simple weighted sum of their outputs, described by

$$COMB_{score}(a) = \alpha ERB_{score}(a) + (1 - \alpha) LSA_{score}(a) \qquad (3)$$

where 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. In our experiments we have also tried to optimize α on the test set, so as to measure the upper bound of our combination method.

Experimental settings

To evaluate our systems, we built nine different benchmark data sets from real exams in Spanish, and an additional one with definitions obtained from Google Glossary (Pérez, Alfonseca, & Rodríguez 2004a). The ten sets are described in Table 1. For each question, we collected a set of students' answers and we asked two different human judges to assign a score to each of them. They also wrote the reference answers for each question. The set of reference answers is the only knowledge source required by ERB, while LSA needs an additional domain specific corpus to be trained.

The common test set of students' answers allows us to perform a comparative analysis of ERB, LSA and their combination. For evaluation purposes, the Pearson's correlation coefficient between the humans' scores and the system's scores is calculated.

To train the LSA system we used the two following corpora:

SA: It is a **small** corpus composed by 1.929 *Student Answers* collected in an Operating Systems course. They have been automatically translated from Spanish to English by using Altavista Babelfish[1]. To preserve a correct evaluation methodology, none of the students' answers contained in this corpus is included in the evaluation datasets.

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*.

Evaluation

In this section we evaluate independently both the ERB and the LSA systems in a common test set, described in the pre-

[1] In a previous work (Alfonseca & Pérez 2004), we have observed that results obtained with ERB do not decrease when using an automatic translation system to port the students' answer to another language. Because the corpus used for training LSA is in English, Altavista Babelfish (*http://world.altavista.com/*) has been used to translate the Spanish training and evaluation set to English in order to make a comparison with the English Ziff-Davis Corpus that is used in the other experiments.

	BLEU		ERB	
SET	**SA-LSA**	**CS-LSA**	**SA-LSA**	**CS-LSA**
1	0.69	0.60	**0.73**	0.62
2	0.40	0.32	**0.54**	0.50
3	**0.25**	0.21	0.22	0.17
4	0.77	**0.79**	0.12	0.37
5	0.50	0.40	**0.68**	0.63
6	0.08	0.05	**0.23**	0.20
7	0.24	0.25	0.31	**0.35**
8	0.36	0.30	**0.42**	**0.42**
9	0.30	0.20	0.77	**0.79**
10	0.55	0.45	**0.87**	0.85
Mean	0.41	0.36	**0.49**	**0.49**

Table 3: Evaluation of the combined systems fixing $\alpha = 0.5$. Cells reports the correlations.

vious section, then we show the benefits of their combination.

The results of this first evaluation is shown in Table 2. LSA has been trained on both the SMALL and the LARGE corpora described in the previous section. With the terms SA-LSA and CS-LSA we will refer respectively to the former and to the latter settings.

ERB is clearly the best "basic" system: it outperforms the original Blue algorithm and it is more accurate than LSA as well. Results also show that the accuracy of LSA improves when the CS corpus is used for unsupervised learning, even though the SA corpus describes the questions domain in much more detail. ERB is also complementary to LSA for most of the questions: ERB achieved the best results just for three questions of ten; and all the systems we compared are *highly uncorrelated*.

The complementarity of ERB and LSA allows us to combine them, adopting the system combination schema described previously. We have tried only the possibilities in which we combined one syntactic approach (Blue or ERB) with one semantic approach (SA-LSA and CS-LSA), discarding the other possibilities. Results are reported in Tables 3 and 4.

Table 3 shows the performances of the systems obtained by combining SA-LSA, CS-LSA, BLEU and ERB. For all the combined systems we fixed α to 0.5, so to assign the average score of the basic modules as a final output. 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 how far we can go with our combination method, we also estimated the best parameter settings, by simply optimizing the parameter α on the test set. The mean correlations of the combined systems are reported in Table 4. In the same table we also report the value of the parameter α exploited to achieve the best results. With this optimization technique, the best combination (ERB and CS-LSA) achieves a correlation of 50% that constitutes the best result measured in our experiments.

	BLEU		ERB	
SET	**SA-LSA**	**CS-LSA**	**SA-LSA**	**CS-LSA**
1	**0.73**	0.61	**0.73**	0.61
2	0.44	0.28	**0.54**	0.38
3	**0.25**	0.09	0.22	0.10
4	0.60	**0.81**	0.12	0.48
5	0.59	0.55	**0.68**	0.64
6	0.13	0.14	**0.23**	**0.23**
7	0.23	0.33	0.31	**0.38**
8	0.43	0.45	0.42	**0.46**
9	0.45	0.62	0.77	**0.81**
10	0.74	0.84	0.87	**0.90**
Mean	0.46	0.47	0.49	**0.50**
Alpha	0.3	0.1	0.5	0.2

Table 4: Evaluation of the combined systems by optimizing the parameter α. Cells reports the mean correlations and the values of α at the bottom.

Even if the difference is not very significant, the external large corpus used to train the CS-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. It means 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 questions 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, in our case, even using a generic corpus (i.e. a corpus that does not include particular references to the questions that we have evaluated) to train the LSA module, the accuracy in the automatic scoring process can be improved.

Conclusions and future work

In this paper we have tested the hypothesis that combining different knowledge sources and algorithms is a viable strategy for an automatic assessment of students' free-text answers. In particular we have presented a combination schema for two different techniques: ERB and LSA. To demonstrate our claim we have compared the performances of each technique in the common experimental framework provided by Atenea. Then we have evaluated their combination.

The results show that, tested as stand-alone modules, ERB outperforms the others. Concerning LSA, using a big corpus slightly improves its accuracy. This allows us to adapt the system to different domains, by simply collecting domain specific documents to train the LSA module.

The combination schema for ERB and LSA has also been found effective: the combinations always perform better than their constituent modules. Although we can obtain slightly better results by optimizing the weights, simply using an equal weight for both systems is also an effective strategy for combination. The mean correlation to the human's scores has reached 50%.

It is important to highlight that none of the modules used requires a deeper linguistic processing than just tokenization and part-of-speech tagging; and the only lexical resources used are the two corpora and the evaluation datasets. This helps in keeping the portability across languages that shallow NLP techniques allow.

This paper opens the following prospective lines:

1. The proposed combinational schema allows us to easily integrate LSA (and possibly many other NLP tools such as anaphora resolution and parsing) inside the general architecture of Atenea. For the future we plan to integrate some of these tools inside Atenea.
2. Furthermore, we believe that the idea of combining syntax and semantics can be further explored, by designing more sophisticated system combination techniques and more complex basic modules for text analysis.
3. We are also interested in following in much more detail the new research direction opened by the use of automatic Machine Translation in the field of CAA. Concretely, we plan to compare the performances of our LSA system when trained on fully monolingual settings with the results reported in this paper, for which the both the students' and the references have been automatically translated, to prove the complete language-independence of the whole procedure. This perspective is very attractive especially to make Atenea helpful for foreign students, that could be allowed to answer the question in their own language.
4. Finally, we plan to go deeper in the direction of applying supervised and unsupervised Machine Learning techniques to the field of CAA, by approaching the task of CAA of free-text answers inside the framework of kernel methods.

Acknowledgments

This work has been sponsored by the Spanish Ministry of Science and Technology, project number TIN2004-03140. We also want to thank to Manuel Alfonseca, Rosa M. Carro and Alvaro Ortigosa for their help in building the students' answer corpus and to Milen Kouylekov for the generic Computer Science corpus.

References

Alfonseca, E., and Pérez, D. 2004. Automatic assessment of short questions with a BLEU-inspired algorithm and a shallow semantic representation. In *Advances in Natural Language Processing*, volume 3230 of *Lecture Notes in Computer Science*. Springer Verlag. 25–35.

Alfonseca, E.; Carro, R.; Freire, M.; Ortigosa, A.; Pérez, D.; and Rodríguez, P. 2004. Educational adaptive hypermedia meets computer assisted assessment. In *Proceedings of the International Workshop of Educational Adaptive Hypermedia, collocated with the Adaptive Hypermedia (AH) Conference*.

Berry, M. 1992. Large-scale sparse singular value computations. *International Journal of Supercomputer Applications* 6(1):13–49.

Deerwester, S.; Dumais, S. T.; Furnas, G. W.; Landauer, T.; and Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science* 41(6):391–407.

Dessus, P.; Lemaire, B.; and Vernier, A. 2000. Free text assessment in a virtual campus. In *Proceedings of the 3rd International Conference on Human System Learning*, 61–75.

Florian, R.; Cucerzan, S.; Schafer, C.; and Yarowsky, D. 2002. Combining classifiers for word sense disambiguation. *Natural Language Engineering* 8(4):327–341.

Foltz, T.; Kintsch, W.; and Landauer, T. 1998. The measurement of textual coherence with latent semantic analysis. *Discourse Processes* 25(2-3). Special Issue: Quantitative Approaches to Semantic Knowledge Representations.

Foltz, P.; Laham, D.; and Landauer, T. 1999. The intelligent essay assessor: Applications to educational technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning* 1(2).

Page, E. 1966. The imminence of grading essays by computer. *Phi Delta Kappan*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W. 2001. BLEU: a method for automatic evaluation of machine translation. Research report, IBM.

Pérez, D.; Alfonseca, E.; and Rodríguez, P. 2004a. Application of the BLEU method for evaluating free-text answers in an e-learning environment. In *Proceedings of the Language Resources and Evaluation Conference (LREC-2004)*.

Pérez, D.; Alfonseca, E.; and Rodríguez, P. 2004b. Upper bounds of the BLEU algorithm applied to assessing student essays. In *Proceedings of the 30th International Association for Educational Assessment (IAEA) Conference*.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys* 34(1):1–47.

Streeter, L.; Pstoka, J.; Laham, D.; and d. MacCuish. 2003. The credible grading machine: Automated essay scoring in the dod. In *Proceedings of Interservice/Industry, Simulation and Education Conference (I/ITSEC)*.

Valenti, S.; Neri, F.; and Cucchiarelli, A. 2003. An overview of current research on automated essay grading. *Journal of Information Technology Education* 2:319–330.