Escuela Politécnica Superior
Departamento de Ingeniería Informática

MATRIX FACTORIZATION MODELS FOR CROSS-DOMAIN RECOMMENDATION: ADDRESSING THE COLD START IN COLLABORATIVE FILTERING

PHD DISSERTATION

Ignacio Fernández Tobías

Madrid, November 2016

PhD thesis title: Matrix factorization models for cross-domain recommendation: Addressing the cold start in collaborative filtering

Author: **Ignacio Fernández Tobías**

Affiliation: Departamento de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid, Spain

Supervisor: **Iván Cantador Gutiérrez**
Universidad Autónoma de Madrid, Spain

Date: November 2016

Committee: **José Ramón Dorronsoro Ibero**
Universidad Autónoma de Madrid, Spain

Pablo Castells Azpilicueta
Universidad Autónoma de Madrid, Spain

Francesco Ricci
Free University of Bozen-Bolzano, Italy

Julio Antonio Gonzalo Arroyo
Universidad Nacional de Educación a Distancia, Spain

Pasquale Lops
University of Bari *Aldo Moro*, Italy

Óscar Corcho García
Universidad Politécnica de Madrid, Spain

Owen Conlan
Trinity College Dublin, Ireland

ABSTRACT

Recommender systems are software tools designed to help users in information access and retrieval tasks. By analyzing previous user interactions with certain information items, these systems estimate the users' preferences (i.e., tastes, interests and needs) for other items to predict and suggest the most relevant ones. Actively investigated since the nineties, recommender systems have gained popularity and sophistication, and nowadays are essential components of numerous business, education, culture and entertainment services, such as e-commerce sites like Amazon.com and eBay, media content providers like Netflix, YouTube and Spotify, and online social networks like Facebook and Twitter.

Multiple recommendation approaches, and remarkably those based on collaborative filtering, have been proposed and successfully implemented over the last years. However, they still have limitations and challenges that in turn represent research opportunities. One of the most notorious of these opportunities is the so called *cold start* problem, which refers to the situation where a new user has recently registered in a system, and for whom there are not enough preferences to deliver relevant personalized recommendations. Two types of approaches have been explored to address the cold start. The first is represented by techniques that intelligently elicit the preferences from the user, while the second includes methods that make use of additional information to infer user preferences. Within this last type of approaches, cross-domain recommendation has recently emerged as a potential solution, exploiting user preferences and item attributes in domains distinct, but related to the target recommendation domain.

Cross-domain recommender systems are currently under research in several fields with particular goals and tasks. In User Modeling, these systems have been proposed as a mechanism to aggregate and mediate user profiles as a cross-system personalization strategy, in Machine Learning they have been explored as a practical application of transfer learning techniques, and in Recommender Systems as a way to mitigate the scarcity of user preference data.

This thesis focuses on the study of cross-domain recommender systems as a solution to the cold start. We first provide an in-depth review of the state of the art in the above mentioned research fields, providing a unifying formalization of the problem, and a categorization of existing approaches and evaluation methodologies. We then present three novel adaptations of the matrix factorization technique for cross-domain collaborative filtering. In particular, we propose a number of models that deal with different sources of infor-

mation, namely social tags, user personality factors, and item semantic annotations. In our experimental work, we empirically evaluate the proposed models on large datasets that span several domains, namely movies, music and books recommendation. The achieved results show that our models are indeed effective in cold start scenarios, not only in terms of recommendation accuracy, but also with respect recommendation novelty and diversity, and domain coverage.

RESUMEN

Los sistemas de recomendación son herramientas software diseñadas para ayudar a los usuarios en tareas de acceso y recuperación de información. Mediante el análisis de interacciones pasadas de usuarios con ciertos ítems de información, estos sistemas estiman las preferencias de los usuarios por otros ítems, con el fin de predecir y sugerir aquellos de mayor relevancia. En investigación activa desde mediados de los noventa, los sistemas de recomendación han ido ganando en popularidad y sofisticación hasta convertirse en parte esencial de numerosos servicios de negocio, educación, cultura y entretenimiento, entre los que se incluyen sitios de comercio electrónico como Amazon.com y eBay, proveedores de contenido multimedia por internet como Netflix, YouTube y Spotify, y redes sociales en línea como Facebook y Twitter.

Múltiples son los métodos de recomendación, y destacables aquellos basadas en filtrado colaborativo, que se han propuesto e implementado con éxito a lo largo de los últimos años. Sin embargo, aún tienen limitaciones y retos particulares, que a su vez suponen oportunidades de investigación. Una de las más notorias de estas oportunidades es la del problema de arranque en frío, o *cold start* en inglés, que se produce cuando un usuario se ha registrado en un sistema recientemente, y para el cual no hay preferencias suficientes con las que poder proporcionar recomendaciones personalizadas relevantes. Dos tipos de aproximaciones se han explorado para tratar el arranque en frío. El primero está representado por técnicas que solicitan preferencias al usuario de manera inteligente, mientras que el segundo incluye métodos que hacen uso de información adicional para inferir tales preferencias. Dentro de este último tipo de soluciones, la recomendación sobre dominios cruzados ha surgido recientemente como una potencial solución, utilizando preferencias de usuario y atributos de ítem en dominios distintos, pero relacionados con el de destino.

Los sistemas de recomendación sobre dominios cruzados son objeto de investigación en varias áreas, con objetivos y tareas diferentes. En Modelado de Usuario se han propuesto como mecanismos de agregación y mediación de perfiles de usuario como estrategia de personalización entre sistemas, en Aprendizaje Automático se han planteado como una aplicación práctica de las técnicas de transferencia de conocimiento, y en Sistemas de Recomendación se han sugerido para mitigar la escasez de preferencias de usuario.

Esta tesis se centra en el estudio las sistemas de recomendación sobre dominios cruzados como solución al problema de arranque en frío. En primer lugar, proporciona una revisión exhaustiva de los tra-

bajos relacionados en las áreas arriba citadas, proponiendo una visión que unifica las existentes formalizaciones del problema, y una categorización tanto de los modelos de recomendación como de las metodologías de evaluación empleadas. Posteriormente presenta una serie de novedosas adaptaciones de la técnica de factorización matricial en filtrado colaborativo para la recomendación sobre dominios cruzados. En particular, propone modelos diseñados para explotar distintas fuentes de datos, a saber, etiquetado social, rasgos de personalidad de los usuarios, y anotaciones semánticas de los ítems. El trabajo experimental llevado a cabo comprende la evaluación empírica de los modelos propuestos sobre grandes colecciones de datos que abarcan múltiples dominios, en particular los de recomendación de películas, música y libros. Los resultados alcanzados muestran la efectividad de los modelos propuestos en situaciones de arranque en frío, en términos no sólo de precisión sino también de novedad y diversidad de las recomendaciones, y de cobertura de los dominios.

ACKNOWLEDGMENTS

This thesis is the result of several years of work, during which I received the support of many people. The following lines are a sign of my gratitude to all of them.

First, I would like to thank my supervisor Iván Cantador for his invaluable advice and guidance during all these years, first as a Master's student and then as a PhD candidate. From him I have learned not only how to do research, but also how to approach problems in general, from complex experiments to never-ending paperwork. He encouraged me during the good times and the not-so-good moments, and was always kind and understanding. His continuous hard work and dedication were decisive in the completion of this thesis, and I feel very lucky to have had him as my mentor and supervisor.

Special thanks also to Pablo Castells for giving me the opportunity to start my path in research, and to the rest of the members of the Information Retrieval Group, Fernando Díez, Pedro Campos, Sergio López, Víctor Villasante, Rocío Cañamares, Javier Sanz-Cruzado, and Sofía Marina Pepa. I would like to specially thank Alejandro Bellogín, Saúl Vargas, and Héctor Menéndez, who preceded me in the pursue of a PhD and from whom I have learned so much. It has been a privilege for me to have shared with them countless chats and discussions.

My gratitude also goes to Francesco Ricci for welcoming me to the Recommender Systems group during my visit to the Free University of Bozen-Bolzano, and also to Matthias Braunhofer, Mehdi Elahi, Mouzhi Ge, Floriano Zini, and David Massimo, with whom I had the pleasure of collaborating. I would also like to thank Peter Mika and Roi Blanco for giving me the opportunity to work with them as an intern in Yahoo! Labs in London, and Fidel Cacheda, Petar Ristoski, and Michael Schuhmacher, whom I met during that time and with whom I shared interesting discussions and foosball games. Finally, I thank Paolo Tomeo and Tommaso Di Noia for the dedication they put in the papers that we did together, and for Paolo's visit to our group during the last winter.

Gracias a los amigos que conocí en la carrera y con los que compartí buenos momentos entre la vorágine de prácticas y exámenes durante cinco, cortos en retrospectiva, años. Mi gratitud también a mis compañeros de piso y a la gente que conocí en Londres y que hicieron aquellos tres meses mucho más acogedores. Gracias en especial a Helena, que ha estado conmigo desde que empecé con esto del doctorado y que me ha aguantado y animado en todo momento. Sin ella habría sido mucho más difícil.

Por último, gracias de manera especial a mi familia, que está detrás de todas las cosas buenas que me han pasado. Discretamente, sin pedir nada a cambio, son los responsables de que yo haya llegado hasta aquí y quienes hacen que haya merecido la pena. A ellos va dedicado este trabajo.

Ignacio Fernández Tobías
November 2016

To achieve great things, two things are needed:
a plan, and not quite enough time.

Leonard Bernstein

A mi familia.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

CONTEXT AND BACKGROUND

1 INTRODUCTION

1.1 MOTIVATION

Information overload on the Web

The last decade has witnessed an exponential increase in the number of resources available in the World Wide Web, most notably since it became widely extended and users started creating and uploading their own content. Reaching millions of users, e-commerce sites like Amazon.com and eBay sell hundreds of millions of products from dozens of categories, multimedia streaming providers like Spotify and Netflix offer access to huge catalogs of music tracks and movies and TV shows, and it is estimated that users upload over 400 hours of video content every minute in YouTube. Besides, online social networks represent most of the activities of the users on the Web, such as Facebook with more than 300 million photos shared every day.

Recommender systems

In this context, the vast and continuously increasing amount of available contents entails an *information overload* problem, since finding relevant information items in huge collections may result in a too time consuming, complex task for humans. Recommender systems are software tools designed to help users in their information access and retrieval tasks. By analyzing the users' previous interactions with certain items, these systems infer the users' preferences for other items to predict and suggest the most relevant ones. They are thus essential components of many business, education, culture and entertainment services.

Cold start

In academia, they have been actively investigated since the nineties, and nowadays represent a consolidated research area, as evidenced by the ACM Conference on Recommender Systems[1], which after 10 editions has become a highly respected international forum. Over the last years, multiple recommendation approaches, and remarkably those based on collaborative filtering, have been proposed and successfully implemented. However, there are still challenges and limitations that offer opportunities for new research. One of the most notorious of these opportunities is the so called *cold start* problem, which refers to the situation where a new user has recently registered in a system, and for whom there are not enough preferences to deliver relevant personalized recommendations. The cold start has received much attention from the research and industry communities, as providing good recommendations for new users is critical to keep them engaged with the system; if suggested items are not relevant, users may perceive the system as not useful, and leave it.

1 ACM Conference on Recommender Systems, RecSys, http://recsys.acm.org

Two main types of solutions have been explored to address the cold start problem. The first is represented by techniques that aim for the intelligent acquisition of user preferences, by directly asking the users to evaluate a limited selection of information items. The second includes methods that make use of auxiliary data to infer user preferences. Within this last type of approaches, cross-domain recommendation has recently emerged as a potential solution, exploiting user preferences and item attributes in domains different, but related to the target recommendations domain.

Cross-domain recommendations

Cross-domain recommendation is an emerging research topic in several fields, each with particular goals and tasks. Works on User Modeling have proposed cross-domain approaches as a mechanism to aggregate and mediate user profiles from different domains as a cross-system personalization strategy. In the Machine Learning field, cross-domain collaborative filtering has been investigated as a practical application of transfer learning techniques, which aim to exploit models learned with datasets having different characteristics and distributions. Finally, in Recommender Systems, cross-domain approaches have been mostly studied as a way to mitigate the scarcity of user preference data in a target domain.

Exhaustive survey on cross-domain recommender systems

This diversity of goals, tasks and approaches has resulted in multiple complementary formulations of the cross-domain recommendation problem. Moreover, there is not a consensus on the definition of recommendation domain, which makes difficult the classification and comparison of the approaches proposed in the literature, and hinders the identification of new research opportunities. A first contribution of this thesis is an in-depth review of the state of the art on cross-domain recommendation in the above fields, providing a unifying formalization of the problem, as well as a categorization of recommendation approaches and evaluation methodologies.

Novel matrix factorization models for cross-domain collaborative filtering

A major issue in the development of cross-domain recommender systems is how to establish a *bridge* between the involved domains in order to support the aggregation or transfer of knowledge from an auxiliary source domain to the target recommendation domain. In this respect, most of the approaches proposed so far focus on collaborative filtering solutions that only exploit user preferences in the form of numeric ratings. This has the advantage of not requiring any other information about the users or the items, which can be highly heterogeneous across domains. It may be, however, a limitation if there are no common users or items between domains. Moreover, as several studies have already shown, auxiliary information about the users or the items content could lead to more effective recommendations.

In this direction, in this thesis we present novel extensions of the matrix factorization method for collaborative filtering (Hu et al., 2008; Koren, 2008). Specifically, we propose matrix factorization models for cross-domain collaborative filtering that aim to mitigate the cold start by exploiting three different sources of information, namely social tags, user personality factors, and item semantic metadata.

Social tags

During the last years, there has been an increasing popularization of social tagging services, in which users upload contents and annotate them with freely chosen words known as *tags*. The set of tags in each system constitutes a collaborative, unstructured knowledge classification scheme that can be considered as a source of user preferences, since users assign tags to own contents and contents they like from others, and thus can be used for recommendation purposes.

Personality factors

Alternatively, personality is a pattern of values, attitudes, and behavioral repertoire that characterizes people, and has certain persistence over life, so that the manifestations of that pattern in different situations have some degree of predictability. In certain domains, it has been shown that people with similar personality traits tend to have similar preferences, which makes personality a potential source of information to provide collaborative filtering recommendations.

Semantic metadata

In addition to collaborative filtering, content-based filtering has been applied in domains where item content and metadata play a key role, either in addition to or instead of explicit ratings and implicit user feedback. With the advent of the Semantic Web, and its reference implementation *Linked Data*, a plethora of structured, interlinked metadata is available on the Web. These metadata also represent a potential source of information to be exploited by content-based and hybrid filtering approaches.

Positive-only feedback as user preferences

Differently to most of previous work in the state of the art, which has focused on the rating prediction task, two of the three proposed approaches are designed to handle positive-only feedback as user preferences for the item ranking task. This is arguably a more realistic scenario, as positive-only feedback (e.g., item click logs, consuming counts, and purchase records) is easily collected implicitly by the system. However, it is often more challenging to handle, since the acquired feedback only serves as evidence of the users' preferences, but does not provide any information about their dislikes.

Results on three domains with large datasets

In our experimental work, we empirically compare the proposed models using large datasets that span several domains, namely movies, music and books recommendations. The achieved results show that our models are indeed effective in cold start scenarios, not only in terms of recommendation accuracy, but also with respect recommendation novelty and diversity, and domain coverage.

1.2 GOALS

Research goals of the thesis

In this thesis we aim to investigate how cross-domain recommendations can be used to mitigate the cold start problem in collaborative filtering. For such purpose, we hypothesize that exploiting auxiliary information, additional to user preferences, about the users and items allows for a more effective transfer of knowledge between domains, and thus the generation of better recommendations in situations of user preference scarcity. With this hypothesis in mind, we state the following specific research goals.

RG1: review the state of the art on cross-domain recommender systems, in order to identify related work addressing the cold start. As mentioned in the previous section, the cross-domain recommendation problem has been approached in several fields, and there is not yet a consensus regarding the formalization of the problem, and a holistic view of the goals and tasks for which the different solutions have been designed. We aim to conduct a rigorous, exhaustive survey to unify perspectives and identify research opportunities.

RG2: develop novel cross-domain recommendation models that exploit auxiliary information in addition to user preferences, and evaluate them rigorously in cold-start situations. Most of the cross-domain approaches proposed so far are based on collaborative filtering methods that only consider user preferences. We, in contrast, propose to exploit other types of information about users and items. The conducted survey of RG1 would let us determine the potential data sources that could benefit cross-domain recommendations in cold start situations. To validate the performance of developed solutions, we require the evaluation of the models to be conducted on several domains and with relatively large datasets. We also require following an evaluation methodology adequate to the cold start. Again, the survey of RG1 would help us on these issues.

RG3: analyze the effectiveness of the proposed cross-domain recommendation models going beyond accuracy. In the literature, the majority of existing cross-domain recommendation approaches has been evaluated in terms of the error in rating predictions. According to the nature of the recommendation models developed in RG2, for this thesis, we aim to address both rating prediction and item ranking tasks, and consider user preferences distinct to numeric ratings. For these reasons, we would evaluate the models by means of both (rating prediction) accuracy and (item ranking) precision metrics, appropriate to either numeric, binary or positive-only feedback. Furthermore, we also aim to analyze the effectiveness of the proposed models according to recommendation properties distinct to accuracy/precision, such as novelty, diversity and coverage. Analogously as for RG2, the literature review from RG1 would allow us to choose the suitable evaluation metrics.

1.3 CONTRIBUTIONS

Main contributions of the thesis

The work done in this thesis has resulted in several contributions to the state of the art on cross-domain recommender systems, which we summarize next.

In Chapter 3 we provide a comprehensive, in-depth review of previous work on cross-domain recommender systems. We **present a formalization of the problem** considering a holistic definition of recommendation domains at different granularity levels in order to **unify notions of domain** considered previously. Moreover, we identify the different tasks addressed in the state of the art, and the pursued recommendation goals, e.g., improving the accuracy of predictions, enhancing user models, and mitigating the cold start. Additionally, we **categorize existing approaches** based on their recommendation technique, distinguishing models that aggregate user preferences and models that link or transfer knowledge from the source domain to the target domain.

In Chapter 4 we present an **extension of matrix factorization that incorporates additional parameters to model and transfer the effect of social tags on the ratings across domains**. We review previous tag-based approaches to cross-domain recommendation in order to identify their strengths and limitations. We take inspiration from a matrix factorization model for single domains that exploits item metadata, and adapt it to the cross-domain scenario. Our approach separately models the contribution of user and item tags, allowing for better capturing their effect on the observed ratings, and computing rating predictions even when the user did not annotate the target items.

In Chapter 5 we present **personality-based matrix factorization models** that exploit information about the users' personality factors to compute recommendations for new users in single- and cross-domain settings. As opposed to numerical ratings, our models are designed to **handle positive-only feedback**. Moreover, we evaluate several **methods to model user personality in matrix factorization**, and provide an **adaptation of the alternating least squares algorithm** to train our models exploiting additional information.

Finally, in Chapter 6 we present three **extensions of matrix factorization for cross-domain collaborative filtering that exploit item semantically-related annotations to bridge domains**. In particular, the semantic information is used to compute inter-domain similarities between the items, which are used to regularize latent item factors. For the three models proposed, we provide **efficient training algorithms to learn the optimal latent parameters** in a dimension at a time, based on a fast version of alternating least squares from the state of the art (Pilászy et al., 2010).

1.4 STRUCTURE OF THE DOCUMENT

Three chapters presenting the solutions proposed in the thesis

As explained in previous sections, in this thesis we present a number of novel matrix factorization models for cross-domain collaborative filtering, which result effective to mitigate the cold start in a target domain, by exploiting auxiliary source domain data distinct to user preferences provided as numeric ratings. In particular, we investigate the exploitation of user-item interactions in the form of social tags, user personality factors, and item semantically related metadata. We will present these solutions in three core chapters of this document.

This variety of the above information sources entails to treat research topics from several fields, such as social tagging data mining in Artificial Intelligence, personality modeling in Psychology, and knowledge representation in the Semantic Web. Furthermore, these types of information have also originated the proposal of particular approaches in the Recommender Systems field, namely social tag-, personality- and semantic-based recommendation methods in both single and cross-domain settings. Taking into account that a very large description of state of the art on all these topics and fields may be unappealing for the reader, specific literature reviews have been distributed in the three corresponding chapters. The three chapters have the same structure, with sections to introduce and motivate the research done, survey existing approaches, present the proposed recommendation models, and report and discuss the results achieved in conducted experiments.

Two chapters analyzing the context and background of the thesis

Although each of such core chapters addresses particular related work, aiming to offer a detailed overview of the context and background of the thesis, in a first part of the document, two chapters have been dedicated to a description of general issues in recommender systems, and an exhaustive survey on cross-domain recommender systems.

Specific content of each chapter

The content of all the chapters is described in more detail next.

Part I: Context and background

- Chapter 2 provides an overview of general issues on **recommender systems**. In the chapter we first formulate the recommendation problem distinguishing the rating prediction and item ranking tasks, explain the main types of user preferences, and discuss the limitation of user preference scarcity in cold start situations, which is main focus of this thesis. We then give a categorization and description of general recommendation techniques, namely content-based and collaborative filtering, and detail matrix factorization collaborative filtering, which is the basis of the recommendation models proposed in the thesis. Finally, we describe methodologies and metrics to evaluate recommender systems, some of which are used in our experimental work.

- Chapter 3 presents a novel, exhaustive survey of the state of the art on **cross-domain recommender systems**. Unifying perspectives from different fields, we first propose a formulation of the cross-domain recommendation problem, tasks and goals. We then propose a categorization of cross-domain recommendation techniques, distinguishing knowledge aggregation methods and knowledge linkage and transfer methods. For each of these types of techniques, we analyze and compare a large number of existing approaches. Analogously to the previous chapter, we conclude with a discussion on issues regarding the evaluation of cross-domain recommender systems.

Part II: Proposed solutions

- Chapter 4 proposes a matrix factorization model for cross-domain collaborative filtering that exploits **social tags** as a source of user preferences that are shared or related between different domains. In the chapter we revise existing social tag-based recommendation approaches for single and cross domains, focusing on the matrix factorization models that have inspired the one proposed in the chapter. Next, we describe the proposed model, and report and discuss empirical results achieved from the evaluation of the model for the rating prediction task in cold start situations, using the well known MovieLens[2] and LibraryThing[3] datasets on the movies and books recommendation domains.

- Chapter 5 proposes matrix factorization models for cross-domain collaborative filtering that consider **user personality factors** as domain-independent features, and exploit them for establishing relationships between user preferences on items from different domains. In the chapter we first motivate the proposed approach by revising previous works that have shown the existence of relationships between personality factors and user preferences in certain domains, and analyzing existing approaches that have incorporated personality information into collaborative filtering heuristics. We then present our personality-based matrix factorization model which, differently to existing works, has been evaluated with large datasets on three domains, namely movies, music and books recommendations. Specifically, we report and discuss empirical results using a dataset extracted from the myPersonality[4] project, which provides a large number of user profiles composed of Facebook[5] *likes* and *Big Five* personal-

2 MovieLens datasets, http://grouplens.org/datasets/movielens
3 LibraryThing dataset, http://www.macle.nl/tud/LT
4 The myPersonality project, http://mypersonality.org
5 Facebook online social network, https://www.facebook.com

ity factor scores. The model is thus evaluated with positive-only feedback, for the item ranking task.

- Chapter 6 proposes matrix factorization models for cross-domain collaborative filtering that, instead of exploiting user-item data in the form of social tags and user-specific data with personality information, focus on the use of **item semantic metadata** to bridge user preferences for items from different domains. In particular, the proposed models make use of semantic-based features and relations automatically extracted from DBpedia[6], which is the structured version of the well known Wikipedia[7] online encyclopedia, and the core knowledge base of the Linked Open Data[8] project. In the chapter we survey state of the art recommendation approaches that exploit Linked Data, and present the proposed matrix factorization models. Similarly to our personality-based models, to evaluate the proposed semantic-based models we conduct experiments for the item ranking task in cold start, with a dataset composed of Facebook *likes*. In this case, liked items are automatically mapped to DBpedia entities, whose metadata are extracted and used to build semantic networks linking items across domains.

- Chapter 7 ends the thesis with overall **conclusions** on the exploitation of information additional to user ratings by matrix factorization models for cross-domain collaborative filtering. In the chapter we also discuss limitations and pending research issues not addressed in the thesis, which may give grounds for further investigation.

1.5 PUBLICATIONS

The work presented in this thesis has resulted in several peer reviewed publications in journals, an edited book, conferences and workshops. We list these publications next, grouped and sorted according to the thesis chapters and research topics with which they are related.

Publications related to Chapter 3, Cross-domain recommender systems

Publications presenting surveys on cross-domain recommender systems

The first contributions of this thesis are the formalization of the cross-domain recommendation problem –unifying perspectives from which it has been addressed–, and the analytical categorization, description and comparison of prior work –conducting an exhaustive survey of a large number of papers in different research areas, namely User Modeling, Machine Learning, and Recommender Systems. The

6 The DBpedia knowledge repository, `http://wiki.dbpedia.org`
7 The Wikipedia online encyclopaedia, `https://www.wikipedia.org`
8 The Linked Open Data project, `http://linkeddata.org`

following book chapter and conference paper present surveys on cross-domain recommender systems with the above contributions:

- Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, Paolo Cremonesi. 2015. **Cross-domain Recommender Systems**. In *Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.), Recommender Systems Handbook - 2nd edition*, pp. 919-959. Springer, ISBN 978-1-4899-7636-9.

- Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskas, Francesco Ricci. 2012. **Cross-domain Recommender Systems: A Survey of the State of the Art**. In *Proceedings of the 2nd Spanish Conference on Information Retrieval (CERI 2012)*, pp. 187-198. Publicaciones de la Universitat Jaume I, ISBN 978-84-8021-860-32.

Publications related to Chapter 4, Social tag-based matrix factorization models for cross-domain collaborative filtering

Publications about social tag-based cross-domain recommendation

In this thesis we empirically validate the hypothesis that social tags can be used to establish relations between user preferences and sentiments for items from different domains, and that such relations can be exploited for recommendation purposes. The proposal and evaluation of social-tag based cross-domain user modeling and matrix factorization collaborative filtering approaches are presented in the following publications:

- Ignacio Fernández-Tobías, Iván Cantador. 2014. **Exploiting Social Tags in Matrix Factorization Models for Cross-domain Collaborative Filtering**. In *Proceedings of the 1st International Workshop on New Trends in Content-based Recommender Systems (CBRecSys 2014)*, pp. 34-41. CEUR Workshop Proceedings 1245, ISSN 1613-0073.

- Ignacio Fernández-Tobías, Iván Cantador, Laura Plaza. 2013. **A Social Tag-based Dimensional Model of Emotions: Building Cross-domain Folksonomies**. *Procesamiento del Lenguaje Natural* 51, pp. 195-202. Sociedad Española de Procesamiento del Lenguaje Natural, ISSN 1135-5948.

- Ignacio Fernández-Tobías, Iván Cantador, Laura Plaza. 2013. **An Emotion Dimensional Model Based on Social Tags: Crossing Folksonomies and Enhancing Recommendations**. In *Proceedings of the 14th International Conference on Electronic Commerce and Web Technologies (EC-WEB 2013)*, pp. 88-100. Lecture Notes in Business Information Processing 152, Springer, ISBN 978-3-642-39877-3.

Publications related to Chapter 5, Exploiting user personality factors in matrix factorization for cross-domain collaborative filtering

Publications about personality-based cross-domain recommendation

In the thesis we exploit existing relationships between personality factors and user preferences for items belonging to different domains, to propose personality-based heuristic methods and matrix factorization models for single- and cross-domain collaborative filtering. These recommendation approaches, together with previous study on the above relationships, appear in the following publications:

- Ignacio Fernández-Tobías, Matthias Braunhofer, Mehdi Elahi, Francesco Ricci, Iván Cantador. 2016. **Alleviating the New User Problem in Collaborative Filtering by Exploiting Personality Information**. *User Modeling and User-adapted Interaction* 26(2), pp. 221-255. Springer, ISSN 0924-1868.

- Ignacio Fernández-Tobías, Iván Cantador. 2015. **On the Use of Cross-Domain User Preferences and Personality Traits in Collaborative Filtering**. In *Proceedings of the 23rd International Conference on User Modeling, Adaptation, and Personalization (UMAP 2015)*, pp. 343-349. Lecture Notes in Computer Science 9146, Springer, ISBN 978-3-319-20266-2.

- Ignacio Fernández-Tobías, Iván Cantador. 2014. **Personality-aware Collaborative Filtering: An Empirical Study in Multiple Domains with Facebook Data**. In *Proceedings of the 15th International Conference on Electronic Commerce and Web Technologies (EC-Web 2014)*, pp. 125-137. Lecture Notes in Business Information Processing 188, Springer, ISBN 978-3-319-10490-4.

- Iván Cantador, Ignacio Fernández-Tobías. 2014. **On the Exploitation of User Personality in Recommender Systems**. In *Proceedings of the 1st International Workshop on Decision Making and Recommender Systems (DMRS 2014)*. CEUR Workshop Proceedings 1278, ISSN 1613-0073.

- Iván Cantador, Ignacio Fernández-Tobías, Alejandro Bellogín. 2013. **Relating Personality Types with User Preferences in Multiple Entertainment Domains**. In *Late-Breaking Results, Project Papers and Workshop Proceedings of the 21st Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*. CEUR Workshop Proceedings 997, ISSN 1613-0073.

Publications related to Chapter 6, Exploiting item metadata in matrix factorization for cross-domain collaborative filtering

Publications about semantic-based cross-domain recommendation

Finally, in the thesis we investigate the exploitation of item metadata to establish inter-domain relationships, and the incorporation of such relationships into cross-domain matrix factorization models for collaborative filtering. The process of extracting item metadata from

Linked Data repositories, the building of multiple-domain semantic networks linking items, and the proposal and evaluation of the above recommendation models are presented in the following publications:

- Ignacio Fernández-Tobías, Paolo Tomeo, Iván Cantador, Tommaso Di Noia, Eugenio Di Sciascio. 2016. **Accuracy and Diversity in Cross-domain Recommendations for Cold-start Users with Positive-only Feedback**. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys 2016)*, pp. 119-122. ACM, ISBN 978-1-4503-4035-9.

- Ignacio Fernández-Tobías, Roi Blanco. 2016. **Memory-based Recommendations of Entities for Web Search Users**. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM 2016)*, pp. 35-44. ACM, ISBN 978-1-4503-4073-1.

- Paolo Tomeo, Ignacio Fernández-Tobías, Tommaso Di Noia, Iván Cantador. 2016. **Exploiting Linked Open Data in Cold-start Recommendations with Positive-only Feedback**. In *Proceedings of the 4th Spanish Conference on Information Retrieval (CERI 2016)*, art. 11. ACM, ISBN 978-1-4503-4141-7.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2014. **Knowledge-based Identification of Music Suited for Places of Interest**. *Journal of Information Technology and Tourism* 14(1), pp. 73-95. Springer, ISSN 1098-3058.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2013. **Ontology-based Identification of Music for Places**. In *Proceedings of the 13th International Conference on Information and Communication Technologies in Tourism (ENTER 2012)*, pp. 436-447. Springer, ISBN 978-3-642-36308-5.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2012. **Knowledge-based Music Retrieval for Places of Interest**. In *Proceedings of the 2nd International Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies (MIRUM 2012)*, pp. 19-24. ACM, ISBN 978-1-4503-1591-3.

- Ignacio Fernández-Tobías, Marius Kaminskas, Iván Cantador, Francesco Ricci. 2011. **A Generic Semantic-based Framework for Cross-domain Recommendation**. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011)*, pp. 25-32. ACM, ISBN 978-1-4503-1027-7.

Other publications related to the thesis

Other publications

The thesis project was defended and evaluated at the Doctoral Consortium of the 21st Conference on User Modeling, Adaptation, and Personalization:

- Ignacio Fernández-Tobías. 2013. **Mining Semantic Data, User Generated Contents, and Contextual Information for Cross-Domain Recommendation**. In *Proceedings of the 21st Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*, pp. 371-375. Lecture Notes in Computer Science 7899, Springer, ISBN 978-3-642-38843-9.

Related with our work on social tag-based recommendation, the following publications present data processing techniques and generated datasets that we further used in our experiments:

- Iván Cantador, Alejandro Bellogín, Ignacio Fernández-Tobías, Sergio López-Hernández. 2011. **Semantic Contextualization of Social Tag-based Item Recommendations**. In *Proceedings of the 12th International Conference on E-Commerce and Web Technologies (EC-Web 2011)*, pp. 101-113. Lecture Notes in Business Information Processing 85, Springer, ISBN 978-3-642-23013-4.

- Ignacio Fernández-Tobías, Iván Cantador, Alejandro Bellogín. 2011. **Semantic Disambiguation and Contextualisation of Social Tags**. In *Advances in User Modeling perspectives - selected papers from UMAP 2011 workshops*, pp. 181-197. Lecture Notes in Computer Science 7138, Springer, ISBN 978-3-642-28508-0.

- Ignacio Fernández-Tobías, Iván Cantador, Alejandro Bellogín. 2011. **cTag: Semantic Contextualisation of Social Tags**. In *Proceedings of the 6th International Workshop on Semantic Adaptive Social Web (SASWeb 2011)*. CEUR Workshop Proceedings 730, ISSN 1613-0073.

2 RECOMMENDER SYSTEMS

Recommender systems are commonly defined as software tools aimed to help users find relevant items —such as products, movies, music and books—, being the relevance of the items often established relying on the users' tastes, interests and needs. They are most useful in situations of very large item catalogs, whose manual exploration is highly time consuming. This is the case of the majority of real-world e-commerce sites, like Amazon.com, which sells millions of products from many different categories, and online media providers, like Netflix, which offers thousands of streaming films and TV series on demand. Recommender systems thus assist the users in decision making processes, saving time and effort by automatically filtering the potentially relevant items. Differently to search engines and other information access services, recommender systems are proactive in their item suggestions, that is, they do not require the user to state her current information needs, as done e.g. through keyword-based queries.

In this chapter we provide an overview of recommender systems. In Section 2.1 we formalize the recommendation problem, describing major recommendation tasks, existing types of user preferences, and the problem of lack and scarcity of user preferences. In Section 2.2 we present the most popular categorization of recommendation techniques, namely *content-based filtering* and *collaborative filtering*, describing popular approaches of each of them. Next, in Section 2.3 we detail matrix factorization for collaborative filtering, the method upon which the cross-domain recommendation models proposed in this thesis will be designed. Finally, in Section 2.4 we describe popular methodologies and metrics in recommender system evaluation, some of which will be used to empirically evaluate the models proposed.

2.1 THE RECOMMENDATION PROBLEM

Formulation of the recommendation problem

The recommendation problem has been approached from numerous perspectives, and has been formulated in diverse ways. In their recent paper, Jannach and Adomavicius, (2016) present a detailed list of purposes a recommender system may have under both the recommendation consumers' and providers' viewpoints, e.g., identifying items that match long-term user preferences, notifying new contents, establishing group consensus, creating additional demand, and increasing user engagement. Considering the most common purpose of helping users find relevant items based on personal preferences, in one of the earliest reviews of the state of the art, Adomavicius and Tuzhilin, (2005) provide the following mathematical formulation of the recommendation problem.

Let $\mathcal{U}$ be the set of users registered in a system, and let $\mathcal{I}$ be the set of items in the system catalog. Let $f : \mathcal{U} \times \mathcal{I} \to \mathcal{R}$, where $\mathcal{R}$ is a totally ordered set, be a *utility function* such that $f(u, i)$ measures how useful item i is for user u. Then, for each $u \in \mathcal{U}$, the goal of a recommender is to find the item $i_u^* \in \mathcal{I}$, not yet known to the user, that maximizes the utility function:

$$i_u^* = \arg\max_{i \in \mathcal{I}} f(u, i) \tag{2.1}$$

The set $\mathcal{R}$ typically refers to positive integers or real numbers within a given range, and quantifies the levels of preference or usefulness items can have for the users.

The most important issue in this formulation is that the utility function f is not defined over the full $\mathcal{U} \times \mathcal{I}$ set, but on a limited subset of it. That is, the utility values are only available to the system for a fraction of the user-item pairs whose interactions were recorded or observed, and the goal of the recommender is to extrapolate f to the whole $\mathcal{U} \times \mathcal{I}$ set by estimating the unknown values of the utility function. Later in Section 2.2 we shall review a number of techniques that perform this estimation from distinct perspectives and exploiting different types of data.

Rating matrix

Throughout the recommender systems literature, the utility function is often represented as a matrix $\mathbf{R}$ of size $|\mathcal{U}| \times |\mathcal{I}|$, commonly denoted as the user-item matrix or the *rating matrix*. In this matrix, rows correspond to users, columns to items, and an entry $R_{ui} \in \mathcal{R}$ contains the value of the utility function $f(u, i)$, which in most cases is missing as f is not fully defined. For the remainder of the chapter, $\mathcal{I}(u) \subseteq \mathcal{I}$ will denote the set of items for which user u expressed explicitly or implicitly a positive/negative preference, i.e., the non-missing entries of the u-th row of $\mathbf{R}$. Likewise, $\mathcal{U}(i) \subseteq \mathcal{U}$ will denote the set of users who have certain preference for item i, i.e., the non-missing entries of the i-th column of $\mathbf{R}$.

2.1.1 *Rating prediction and item ranking tasks*

As already formulated, the ultimate goal of a recommender system is to identify the most useful items for each user. This is generally addressed as two tasks, namely (i) predicting the values of unknown ratings to suggest the items with highest estimated ratings, and (ii) directly generating and suggesting a ranking of items.

Rating prediction task

Early works in the field focused on the *rating prediction* task, aiming to accurately estimate the users' missing ratings, and recommend the items with the highest rating estimations. Hence, the addressed problem is to generate predictions $\hat{r}(u, i)$ as close as possible to the true ratings r_{ui}.

Until recently, the rating prediction task has been the most popular way to address the recommendation problem, likely encouraged by the availability of datasets containing user preferences in the form of ratings, and events like the Netflix Prize[1] competition, where researchers and practitioners competed for a $1 million prize, awarded to those who developed a recommendation algorithm that improved the rating prediction error by 10% with respect to that of the Netflix algorithm. Remarkably, the trend has changed in the research community in the last years, as experimental evidence has shown that more accurate rating predictions do not necessarily correlate with higher user satisfaction (Cremonesi et al., 2011a; McNee et al., 2006). From a practical point of view, it could also be argued that accuracy only matters for items with high estimated ratings, since the remainder are not likely to be recommended to the user, who is only interested in a limited list of suggestions. Additionally, from an algorithmic perspective, the fact that ratings are missing not at random causes most rating prediction models to generate biased estimations (Steck, 2010).

Item ranking task

As opposed to rating prediction, the *item ranking* task –also known as the *top*-N *recommendation* task— aims to directly generate an ordered list of the items that are most likely to be of interest to the user. This is arguably a more natural scenario, since the user expects the system to automatically deliver a list of the N most relevant items in the catalog (Cremonesi et al., 2010). The task can be formulated as follows. Let u be the target user, and $\mathcal{C} \subseteq \mathcal{I}$ the set of candidate items for recommendation. The goal of the recommender is to generate a ranking of the candidate items $R(u) \subseteq \mathcal{C}$ such that $|R(u)| \leqslant N$, sorted by decreasing estimated relevance for u. This ranking is usually built by computing scores $s(u, i)$, which act as a proxy for the unknown true relevance of each candidate item, and are sorted decreasingly. Hence, in contrast to rating prediction, item ranking is not restricted to explicit numeric ratings, and can be applied to implicit positive-only user feedback. Still, an item ranking for numeric ratings can be generated simply using the rating estimations as relevance scores, i.e. $s(u, i) = \hat{r}(u, i)$.

For the item ranking task, rather than computing an individual score for each candidate item, a new array of *Learning to Rank* techniques apply machine learning algorithms to directly learn the optimal permutation of the candidate items (Karatzoglou et al., 2013). By optimizing ranking-based metrics commonly used in information retrieval evaluations, these techniques have been shown to achieve better results than score-based approaches (Shi et al., 2012).

1 The Netflix prize competition, http://www.netflixprize.com

2.1.2 *Sources of user preferences*

In order to generate personalized suggestions of items, recommender systems need records of previous item choices and preferences as evidence of the users' tastes and interests, i.e., they need enough user-item interactions to accurately extrapolate the utility function. This feedback can be explicitly stated by the users, or implicitly inferred by the system.

Explicit feedback

Explicit feedback refers to interactions that the system gathers by directly asking the users to provide evaluations about items they know. This evaluations are usually collected in the form of *ratings*, numerical values within a specific range that quantitatively measure the degree of user preference for a given item. Usually, ratings are represented graphically as *stars*, and are internally stored as integers in a limited scale such as $1-5$, where a 1-star rating means total dislike, and a 5-star rating means total like. Other forms of explicit feedback include thumbs up/down or like/dislike evaluations.

Through explicit feedback the users manually evaluate items, having more control over the information the system uses for generating personalized recommendations. On the negative side, the users have to consciously provide their preferences, which requires time and effort they may not be willing to take. Moreover, explicitly collected interactions in real-world applications tend to be biased to positive feedback, as users are more likely to rate items they like, which results in rating distributions skewed towards high values, and algorithms misleading rating predictions (Marlin and Zemel, 2009).

Implicit feedback

Instead of expecting the users to proactively provide information about their preferences, some systems automatically record the users' interactions with the items as a source of preferences. This information is commonly referred as *implicit feedback*, and usually consists of item click logs, browsing sessions, consuming counts, and purchase records. Implicit feedback is easier to obtain than explicit feedback. However, it has some characteristics that recommendation models should take into account (Hu et al., 2008):

- Implicit feedback is *positive-only*, i.e., the observed interactions provide a hint of what items a user likes, but give no information about her dislikes. A missing user-item interaction can be due to the actual dislike from the user, who chose not to interact with the item, but can also result from the user's ignorance of the item.

- Feedback collected implicitly tends to be noisy, as it may not be a clear evidence of actual user preferences. For instance, a user may click and watch a recommended movie that she later dislikes, or purchase a product in an e-commerce site as a gift for someone else.

- As opposed to explicit ratings, whose values represent degrees of user preference, the number of registered interactions indicates confidence about the user's preferences. For example, the more times a user listens to a music track, the stronger the evidence the user actually likes it.

In order to deal with implicit feedback, some early approaches utilized heuristics to transform user-item interaction counts to explicit ratings, e.g., Celma, (2010) mapped music listening counts to ratings on a $[1-5]$ scale binning the listening cumulative distribution. Parra et al., (2011) presented a mixed-effects logistic regression model to compare the performance of recommendations based on implicit versus explicit feedback.

2.1.3 *User preference scarcity and the cold start*

The different recommendation techniques have particular advantages and disadvantages, mainly due to the fact that they exploit different types of information. Some challenges, however, are inherent to approaches due to the characteristics of the datasets used for making recommendations. Among them, in this section we introduce the general situation of user preference scarcity, which entails the *rating sparsity* and *cold start* problems. We leave for Section 2.2 particular limitations of each type of recommendation technique.

Rating sparsity

The *rating sparsity* problem refers to the fact that the amount of available user-item interactions (ratings) is very small compared to the number of possible user-item pairs, i.e. most of the entries in the rating matrix **R** are missing. Sparsity is quantitatively measured as

$$\mathtt{sparsity}(\%) = 100\left(1 - \frac{|\mathcal{R}|}{|\mathcal{U}| \cdot |\mathcal{I}|}\right) \tag{2.2}$$

In general, datasets in real-world applications are extremely sparse. For instance, the dataset released for the Netflix prize competition contains over 100 million ratings, but its sparsity level is over 98%. Additionally, most of the ratings tend to be concentrated on a small number of very active users and popular items, resulting in highly skewed power-law user preference distributions. As a consequence, recommendation algorithms may tend to focus on the dense regions of the rating matrix, and struggle predicting for users and items in the long tail.

Cold start

The *(user) cold start* problem, on the other hand, arises when a new user registers into the system, and has not yet provided any preference feedback, either implicit or explicit, or when the user has interacted with the system, but the number of collected preferences is not enough to build an accurate user profile with which the system can compute reliable recommendations. The cold start has been mainly addressed from two perspectives. The first perspective corresponds to active learning techniques (Elahi et al., 2016), which attempt to collect feedback by directly asking the user to rate certain items before generating the recommendations. These techniques usually seek popular items that the user is likely to know, and whose rating would be useful to improve the overall system performance. The second perspective is based on the exploitation of additional side information about the user in the recommendation process. For instance, Pazzani, (1999) used demographic data like gender, age, area code, education and employment information, to compute user-user similarities, and Braunhofer et al., (2015c) showed that information about the user's personality can be more effective in some applications. Recently, cross-domain recommendation methods have been proposed that exploit user preferences in different, source domains to mitigate the lack of information in the target domain (Cantador et al., 2015). In this thesis we investigate the use of matrix factorization models in cross-domain collaborative filtering, addressing the cold start problem by mining different sources of user and item information, namely social tags, personality factors, and semantic annotations.

2.2 CATEGORIZATION OF RECOMMENDATION TECHNIQUES

As the research on recommender systems has been progressively gaining momentum, numerous and diverse recommendation approaches have been proposed. The different approaches can be categorized based on various criteria, such as the task they target, namely rating prediction and item ranking, and the type of user preferences they handle, i.e., implicit and explicit feedback.

Memory- and model-based recommendation methods

From the algorithmic point of view, recommendation techniques can be classified into memory- and model-based methods depending on the underlying type of algorithm to estimate the relevance of the items for the target user. Memory-based methods rely on heuristics to directly estimate item relevance (Cantador et al., 2010b; Herlocker et al., 1999; Resnick et al., 1994; Sarwar et al., 2001). Because of their ad-hoc nature, these methods are usually easier to implement and tune, but are less flexible in the sense that the data they use have to satisfy the assumptions upon which the methods are built in order to achieve good performance. Model-based methods, in contrast, use machine learning techniques to build a relevance prediction model from the data. Examples of model-based models include artificial neural networks (Salakhutdinov et al., 2007), Bayesian networks (Breese et al., 1998; Campos et al., 2010), and latent factor models (Hofmann, 2004; Koren et al., 2009; Lee et al., 2001), among others. Most of these methods encode a set of assumptions about the generative process of the observed feedback by introducing variables and parameters aimed to explain the user-item interactions. The optimal values for such parameters are learned in a training phase, by minimizing a suitable loss function over the predictions of the model and the actual user preferences. As a consequence, model-based approaches have high flexibility to better explain the observed user preferences, but may suffer from overfitting and other well-known issues in machine learning practice. Moreover, another disadvantage of model-based methods is their lack of interpretability, making it difficult to extract conclusions about the data, and insights from the learned model.

Content-based filtering and collaborative filtering

In addition to the above categorizations, one the most popular criteria used in the literature to categorize recommendation techniques is based on the type of information they exploit about the user-item interactions to compute item relevance. Specifically, recommendation approaches are generally classified as follows:

- *Content-based filtering* approaches, which analyze and exploit content features and characteristics of the items, to suggest the items that are most similar to the ones the user liked in the past.
- *Collaborative filtering* approaches, which disregard content information and only leverage patterns of user-item interactions (ratings), so that the user is suggested with items preferred by other users with "similar" preferences.
- Other types of approaches include *hybrid filtering* methods, which combine the previous two types of information, *social-based recommendation* methods, which exploit user connections in social networks to suggest items liked by friends or other trusted users, and *context-aware recommendation* methods, which take into account the user's context (e.g., current location and time, weather conditions, mood) to deliver more relevant recommendations.

In the next subsections we review representative work for each of these types or recommendation approaches.

2.2.1 *Content-based filtering approaches*

Content-based (CB) filtering methods (Lops et al., 2011) recommend items with contents similar to those of items preferred by the user in the past. Hence, a main issue in this type of information filtering is modeling item content in a suitable fashion for its automatic analysis and further exploitation.

Vector Space Model

Many CB approaches are based on the well-known Vector Space Model (VSM) (Baeza-Yates and Ribeiro-Neto, 1999) for ad-hoc document filtering used in the Information Retrieval field. According to the VSM, each item is represented as a vector of features $\vec{v}_i = (w_1, ..., w_N) \in \mathbb{R}^N$, where the value of w_j indicates the importance of the j-th feature to describe the item, and N is the number of considered features. In the context of document filtering, features are usually keywords extracted from text, typically after some preprocessing steps to remove uninformative terms. More generally, features can be any characteristic useful to describe the content of an item. As an example, in the case of movie recommendations, features to be considered may include the director of the movies, their cast, genres such as *action* or *comedy*, year of release, and so on.

There are multiple alternatives to determine the *weight* or importance of the features. Among them, TF-IDF can be considered as the most commonly used. In this technique, the weight of a feature f for item i is computed as a combination of two factors, namely the *term frequency* (TF) and the *inverse document frequency* (IDF):

$$\mathrm{TFIDF}(f, i) = \mathrm{TF}(f, i) \cdot \mathrm{IDF}(f) \tag{2.3}$$

The TF factor counts the number of times f is associated with i, which can be binary to model the presence or absence of such feature, e.g., the participation of certain actor in a movie, or an integer, e.g., the number of occurrences of a term within a text document. In this context, it is usual to normalize the TF values as follows:

$$\mathrm{TF}(f, i) = \frac{\mathrm{count}(f, i)}{\max_{f'} \mathrm{count}(f', i)} \tag{2.4}$$

The intuition behind TF is that a given feature is more likely to be relevant for an item as more it is used in the representation of the item. The IDF factor, on the other hand, captures the discriminative power of each feature across the whole collection of items, and is computed as:

$$\mathrm{IDF}(f) = \log \frac{|I|}{n_f} \tag{2.5}$$

where n_f is the number of items in $\mathcal{I}$ that contain feature f. The more common a feature is in the item collection, the larger n_f, and the less informative f is to characterize i, which results in a small IDF value. In addition to TF-IDF, other popular choices like BM25 have been shown to be effective for recommendation (Cantador et al., 2010b).

Memory-based approaches to content-based filtering

Once the item contents are modeled in terms of feature vectors, a user's profile $\vec{v}_u \in \mathbb{R}^N$ can be defined by aggregating the models of the items that the user liked, e.g., by averaging the corresponding item vectors. Then, the utility of item i for user u is heuristically estimated by means of similarity metrics of the corresponding feature vectors, a popular example of which is the cosine similarity:

$$\cos(\vec{v}_u, \vec{v}_i) = \frac{\langle \vec{v}_u, \vec{v}_i \rangle}{\|\vec{v}_u\| \cdot \|\vec{v}_i\|} \tag{2.6}$$

Model-based approaches to content-based filtering

In this context, machine learning techniques have also been applied for CB recommendation. For example, Pazzani and Billsus, (1997) use a Naive Bayes model to classify items as relevant or not relevant, as explained next. Let i be an item with features $f_1, ..., f_N$. Using Bayes' rule and the naive independence approximation, the probability of i belonging to class C (relevant vs. not relevant) is estimated as:

$$p(C|i) = p(C|f_1, ..., f_N) \propto p(C)p(f_1, ..., f_N|C) = p(C)\prod_{j=1}^{N} p(f_j|C) \quad (2.7)$$

where $p(C)$ is the prior probability of class C, and $p(f_j|C)$ are the probabilities of observing features f_j among the items of class C, all of which are automatically learned from training data. Finally, a ranking of candidate items can be generated by sorting them by decreasing probability of belonging to the relevant class.

Limitations of content-based filtering

Despite their wide use, content-based approaches suffer from several limitations. First, with the exception of text documents, it is not straightforward how to automatically extract meaningful content features from the items, and expert domain knowledge is required to avoid *limited content analysis*. A second issue is known as *over-specialization*, which means that CB systems tend to recommend items that are too similar to those in the user profile. This lack of novelty may be contrary to the user's desire to discover new items, as usually expected from a recommender system. Finally, CB systems struggle to handle cold start situations, as the user profiles built from a few items do not accurately represent the users' preferences.

2.2.2 *Collaborative filtering approaches*

Collaborative Filtering (CF) methods (Herlocker et al., 1999; Koren and Bell, 2015) exploit user-item (rating) interactions to estimate the utility of the items. One advantage of CF methods is that no content information is needed, since only rating patterns are used. Furthermore, by leveraging ratings from users distinct to the target user, CF is able to recommend potentially more novel items, partially avoiding the above mentioned problem of over-specialization.

User-based kNN for numeric ratings

Two of the most popular approaches to CF are based on the k-nearest neighbors (kNN) heuristic (Ning et al., 2015), namely *user-based kNN* and *item-based kNN*. *User-based kNN* (Herlocker et al., 1999; Resnick et al., 1994) recommends a user with items preferred in the past by like-minded users, where the degree of similarity between two users is measured in terms of certain rating-based metric. In the rating prediction task, a common choice for such similarity metric is the Pearson's correlation coefficient between the ratings of users u and v:

$$sim_{PC}(u, v) = \frac{\sum_i (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_i (r_{ui} - \bar{r}_u)^2}\sqrt{\sum_i (r_{ui} - \bar{r}_u)^2}} \quad (2.8)$$

where $\bar{r}_u$ and $\bar{r}_v$ are the average ratings of u and v, respectively. In order to compute the predictions for the target user, usually only the ratings of the most similar users –the *neighborhood* $N(u)$– are taken into account. The most common option is to select the k users with highest similarity as neighbors, although other alternatives are possible, such as picking those users with similarity value over a fixed threshold. Finally, the rating of user u for item i is estimated by interpolation of the neighbors' ratings:

$$\hat{r}(u,i) = \bar{r}_u + \frac{1}{C} \sum_{v \in N(u)} \mathrm{sim}(u,v)(r_{vi} - \bar{r}_v) \cdot \mathbb{1}(i \in I(v)) \tag{2.9}$$

The normalization constant $C = \sum_{w \in N(u)} \mathrm{sim}(u,w)$ is set so that rating predictions are computed within a fixed range. The average ratings $\bar{r}_u$ and $\bar{r}_v$ are included to account for the biases of different users, as some people tend to record higher ratings than others. The indicator function $\mathbb{1}(p)$ is set to 1 *iff* the predicate p is true, and 0 otherwise. This has the effect that only those neighbors that rated the target item i contribute to the item relevance prediction. This definition of the neighborhood may result in null relevance predictions if the target item was rated by none of the neighbors. To avoid this, item-targeted neighborhoods $N_i(u)$ with the most similar users *that rated item* i should be considered. However, such neighborhoods come with a significant increase in computational cost, as a particular neighborhood has to be built for each user-item pair. Therefore, in practice it is usual to use a single neighborhood $N(u)$ for all item relevance predictions of user u regardless the target item, at the cost of lower coverage of potential items to be recommended.

User-based kNN for positive-only feedback

In presence of positive-only feedback, as opposed to numeric ratings, it is usual to use set-based metrics for computing the similarity between users. In this scenario, two users are regarded similar if they interacted with similar items, as the evidence of an interaction implicitly conveys a positive preference. Popular metrics for user similarity are the Jaccard's coefficient and the cosine metric:

$$\mathrm{sim}_{\mathrm{Jaccard}}(u,v) = \frac{|I(u) \cap I(v)|}{|I(u) \cup I(v)|} \tag{2.10}$$

$$\mathrm{sim}_{\cos}(u,v) = \frac{|I(u) \cap I(v)|}{\sqrt{|I(u)||I(u)|}} \tag{2.11}$$

When dealing with positive-only feedback, we aim to build a ranking of recommended items, and thus predict a score $s(u,i)$ for sorting the candidate items, rather than computing rating predictions as in Equation 2.9. Specifically, the score of candidate item i for user u is

$$s(u,i) = \sum_{v \in N(u)} \mathrm{sim}(u,v) \cdot \mathbb{1}(i \in I(v)), \tag{2.12}$$

with the same consideration as before about the item-specific neighborhoods. Two differences are worth noticing with respect to Equation 2.9. First, the average ratings $\bar{r}_u$ and $\bar{r}_v$ are omitted as we no longer deal with ratings but with positive-only feedback. Second, the the score $s(u,i)$ is only used to sort the items in the recommended ranking, and thus does not need to be normalized, hence the omission of the normalization constant C.

Item-based kNN for numeric ratings

The *item-based kNN* method (Sarwar et al., 2001), instead of recommending items liked by other users with similar tastes, works in a dual fashion by suggesting items similar to those the user liked in the past. The key difference with respect to content-based methods relies on how the item similarities are computed. Rather than analyzing the content characteristics of the items, item-based kNN approaches exploit rating patterns, determining that two items are similar if they are frequently preferred together by the users. That is, two items are similar if they are liked by similar sets of users. Analogously to the user-based method, Pearson's correlation coefficient is a common choice for computing item-item similarities,

$$\mathrm{sim}_{PC}(i,j) = \frac{\sum_u (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_u (r_{ui} - \bar{r}_i)^2}\sqrt{\sum_u (r_{uj} - \bar{r}_j)^2}} \tag{2.13}$$

and ratings are predicted as

$$\hat{r}(u,i) = \bar{r}_i + \frac{1}{C} \sum_{j \in N(i)} \mathrm{sim}(i,j)(r_{uj} - \bar{r}_j) \cdot \mathbb{1}(j \in I(u)) \tag{2.14}$$

where the neighborhood $N(i)$ is built with the k most similar items to i or with the items with a similarity score upon certain threshold. In practice, however, only the items rated by the target user u will contribute to the prediction, and it is common to replace the neighborhood $N(i)$ with the user's rated items $I(u)$.

Item-based kNN for positive-only feedback

The formulation for positive-only feedback is also analogous to the user-based method. Item similarities are computed on the basis of the sets of users that rated each, e.g., using Jaccard's coefficient or the cosine metric:

$$\mathrm{sim}_{Jaccard}(i,j) = \frac{|U(i) \cap U(j)|}{|U(i) \cup U(j)|} \tag{2.15}$$

$$\mathrm{sim}_{cos}(i,j) = \frac{|U(i) \cap U(j)|}{\sqrt{|U(i)||U(j)|}} \tag{2.16}$$

The score for raking the items disregards rating averages and normalization factors, and is computed as follows:

$$s(u,i) = \sum_{j \in I(u)} \mathrm{sim}(i,j) \tag{2.17}$$

As it is common in practice, here we have replaced the neighborhood $N(i)$ with the set of the user's preferred items $I(u)$, in order to illustrate the above mentioned discussion.

Due to its flexibility, the nearest-neighbors framework for collaborative filtering has been extensively studied in the literature, including alternative strategies for selecting the neighbors (Bellogín and Parapar, 2012; Xue et al., 2005), other similarity measures (Ma et al., 2007), and even inverting the neighborhoods recommending users to items in order to increase sales diversity (Vargas and Castells, 2014). kNN methods have also been successfully implemented in the industry, such as at Amazon.com (Linden et al., 2003).

Model-based approaches to collaborative filtering

Regarding model-based methods, various machine learning techniques have been applied to the CF task. Examples include Restricted Boltzmann Machines (Salakhutdinov et al., 2007), Gaussian Processes (Lawrence and Urtasun, 2009), and latent factor models (Blei et al., 2003; Hofmann, 2004; Koren et al., 2009). Latent factor models describe statistical models with hidden variables that explain the generative process of the observed data, such as the Probabilistic Latent Semantic Analysis of Hoffman et al. Hofmann, (2004), Latent Dirichlet Allocation (Blei et al., 2003), and Matrix Factorization (Koren et al., 2009). The latter includes some of the most popular methods for model-based Collaborative Filtering, which we review in detail in Section 2.3.

Limitations of collaborative filtering

Despite their success, CF approaches are known to lose effectiveness when the available data is very sparse, as their predictions are solely computed on the basis of observed interactions. In particular, CF systems are not able to deal with the *new item problem*. If an item is added to the system catalog, it needs to be rated by a number of users in order to be eligible for recommendation. Content-based approaches, in contrast, are able to recommend new items by exploiting their content information, even if they have few or no interactions. Additionally, if a user has somewhat *unique* preferences, CF methods will struggle to find similar users and will not be able to provide good recommendations (Adomavicius and Tuzhilin, 2005).

2.2.3 *Other types of recommendation approaches*

Hybrid filtering

In order to overcome particular limitations of both content-based and collaborative filtering, *hybrid filtering* methods have been proposed that combine rating data with other information, such as item content features and user demographics (Pazzani, 1999). According to Adomavicius and Tuzhilin, (2005), hybrid recommendation approaches can be classified in terms of how the above combination is performed, as follows:

- Combining predictions of separate recommenders: the recommendations of content-based and collaborative filtering methods are combined into a single item ranking, e.g., using a voting scheme (Pazzani, 1999) or a linear combination of relevance scores (Claypool et al., 1999).

- Injecting content-based information into the CF framework, for instance by computing user similarities based on content-based profiles rather than common rating patterns (Pazzani, 1999).
- Adding collaborative characteristics to content-based methods, such as matrix factorization on content-based profiles (Soboroff and Nicholas, 1999).
- Combining content-based and collaborative rating information into a single recommendation model, e.g., the work of Barjasteh et al., (2015), where the matrix factorization model is extended by transferring latent features extracted from user or item side-information matrices.

Social-based recommender systems

Additionally, a recent research trend in recommender systems explores how to exploit social network information in collaborative filtering. Broadly, *social-based recommendation* methods suggest items that the target user's friends liked. For instance, Liu and Lee, (2010) adapt the user-based kNN approach computing the neighborhood $N(u)$ as the set of user u's friends in the social network, and Bellogín et al., (2013) propose an approach that recommends the most popular items among the user's friends. An advantage of these methods is their ability to cope with the user cold start problem and the easy interpretability of generated recommendations.

Extensive research has also been conducted on model-based methods that extend the matrix factorization framework with social information. Ma et al., (2008) propose an algorithm that jointly factorizes the user-item rating matrix and the user-user social interaction matrix, showing that the social information leads to more accurate rating predictions. In a later work (Ma et al., 2009), the authors present an approach that computes rating predictions by averaging the target user's prediction and her friend's predictions, also in the context of matrix factorization. Finally, the work by Jamali and Ester, (2010) extends MF with social regularization, so that the model learns that friends in the social network should have similar latent features.

Context-aware recommender systems

All the previous approaches are focused on recommending items to the users without considering the context in which the user-item interactions will take place, which can be crucial in some applications. For instance, the utility of a movie recommendation may vary significantly depending on whether the user will watch a recommended movie alone or with family or friends. *Context-aware recommendation* methods (Adomavicius and Tuzhilin, 2015) extend the recommendation space with additional contextual dimensions, which capture information such as the time of the day, the day of the week, the user's location, the weather conditions, and the user's mood. Adomavicius and Tuzhilin, (2015) classify context-aware approaches according to the stage in the recommendation process where the contextual information is exploited:

- Pre-filtering contextual approaches retain the user-item interactions that match the target user's context, and use them by mens of standard recommendation algorithms (Adomavicius et al., 2005; Baltrunas and Ricci, 2009; Codina et al., 2013).

- Post-filtering contextual approaches take the output of a standard recommendation algorithm, and adjust the resulting recommendation list using contextual information (Panniello et al., 2009).

- Contextual modeling approaches directly exploit the context information within the relevance prediction process, e.g., using Support Vector Machines (Oku et al., 2006) or matrix factorization techniques (Baltrunas et al., 2011).

The area of context-aware recommender systems is very active, and is plenty of research opportunities. In particular, the adoption of smartphones, which include a variety of sensors that let capture the user's context, presents new challenges such as the proactivity of recommendations, i.e. determining the best moment to present the user with a recommendation (Braunhofer et al., 2015a), and the automatic detection of the relevant contextual signals for particular recommendation domains (Braunhofer et al., 2015b).

2.3 MATRIX FACTORIZATION COLLABORATIVE FILTERING

Reduction of the rating matrix dimensionality

Matrix Factorization (MF) models are among the most popular approaches to collaborative filtering, and have been actively investigated since they were introduced in the context of the Netflix prize competition (Bell and Koren, 2007a). As opposed to the user- and item-based algorithms discussed in Section 2.2.2 that use heuristics for their item relevance estimations, MF methods train a statistical model from the available data using machine learning techniques. Specifically, they perform a *dimensionality reduction* of the highly sparse rating matrix into a subspace of latent factors, which aim to capture implicit properties of users and items. In order for MF to be effective, the dimension k of the latent subspace is assumed to be much smaller than the number of users and items, $k \ll \min(|U|, |I|)$, essentially acting as a *bottleneck* that compresses the sparse input while retaining enough information to explain the observed user-item interactions.

2.3.1 *Matrix factorization models for rating prediction*

LSA and SVD

Recommendation models based on MF have their roots on the *Latent Semantic Analysis* (LSA) technique (Deerwester et al., 1990), widely used in Natural Language Processing and Information Retrieval. LSA attempts to automatically infer concepts implicit in text documents by approximating the term-document matrix with a truncated Singular Value Decomposition (SVD) of lower rank. The first MF approaches for recommendation borrowed the same idea, and applied it to the user-item matrix in the rating prediction task (Sarwar et al., 2000). In contrast to LSA, the SVD is not well defined for sparse matrices as those commonly found in recommender systems, and hence the above approaches relied on imputation techniques to fill the missing matrix entries before applying SVD.

User and item latent features

Rather than filling the rating matrix, which may introduce inaccurate information, subsequent approaches aimed to only factorize observed ratings instead of the whole matrix. One of the first and most popular methods in this line is the model proposed by Funk, (2006), in which each user u is assigned a vector $\vec{p}_u \in \mathbb{R}^k$ of latent features automatically inferred from the data, and similarly each item i is assigned a vector $\vec{q}_i \in \mathbb{R}^k$ in the same subspace. Intuitively, latent features aim to capture properties implicit in the data —such as the amount of *comedy* or *action* in the case of movies—, but does not need to be interpretable at all, as this is not enforced in the model (Koren and Bell, 2015). Ratings are then estimated as the dot product of latent feature vectors:

$$\hat{r}(u,i) = \langle \vec{p}_u, \vec{q}_i \rangle \tag{2.18}$$

Latent factor matrices

Equivalently, the rating matrix $\mathbf{R}$ is factorized as $\mathbf{R} \approx \mathbf{P}\mathbf{Q}^\top$, where $\mathbf{P}$ is a $|U| \times k$ matrix with the user vectors $\vec{p}_u$ as rows, and respectively $\mathbf{Q}$ is $|I| \times k$ contains the $\vec{q}_i$ as rows. The values of these matrices are automatically estimated from the data, by minimizing the Mean Squared Error of the ratings predicted against the ratings observed in a training set. That is, $\mathbf{P}$ and $\mathbf{Q}$ are chosen to minimize to following loss function:

$$\mathcal{L}(\mathbf{P},\mathbf{Q}) = \sum_{(u,i)\in\mathcal{R}} (r_{ui} - \langle \vec{p}_u, \vec{q}_i \rangle)^2 + \lambda \left(\|\vec{p}_u\|^2 + \|\vec{q}_i\|^2 \right) \tag{2.19}$$

where $\mathcal{R}$ is the set of observed ratings, i.e. the set of non-zero entries of the rating matrix $\mathbf{R}$, and $\lambda > 0$ is a regularization hyper-parameter used to prevent overfitting.

Stochastic Gradient Descent

In (Funk, 2006) this function is minimized using *Stochastic Gradient Descent*, a widely used optimization technique that iteratively updates the parameters in the opposite direction of the gradient. When applied to Equation 2.19, this technique yields the following update rules for the parameters $\vec{p}_u$ and $\vec{q}_i$ for each rating r_{ui} in the training set:

$$\vec{p}_u \leftarrow \vec{p}_u - \eta \left(e_{ui}\vec{q}_i + \lambda\vec{p}_u \right) \tag{2.20}$$
$$\vec{q}_i \leftarrow \vec{q}_i - \eta \left(e_{ui}\vec{p}_u + \lambda\vec{q}_i \right) \tag{2.21}$$

The *learning rate* η is a hyper-parameter that controls the extent to which the model parameters are updated in each iteration, and is carefully chosen; too large values may make the algorithm fail to converge, while too small values may make its convergence very slow. e_{ui} is the prediction error, and is defined as $e_{ui} \triangleq r_{ui} - \hat{r}(u, i)$.

Alternating Least Squares

In addition to Stochastic Gradient Descent, other optimization techniques have been explored in the literature, such as *Alternating Least Squares* (Bell and Koren, 2007b), which is the standard technique followed in MF models for positive-only feedback (Section 2.3.2).

The basic SVD model by Funk, (2006) is easily extensible, and has served as a building block for more complex matrix factorization models. For instance, Koren, (2008) proposed the SVD++ model, which includes additional parameters to account for implicit feedback in rating predictions. Further extensions of SVD introduce temporal variables to capture the evolution of user preferences through time (Koren and Bell, 2015).

2.3.2 *Matrix factorization models for positive-only feedback*

The core ideas behind the standard Matrix Factorization model for collaborative filtering have also been applied to the item ranking task when positive-only feedback is available instead of numeric ratings. Recommendation models designed for this type of data must take into account its particular characteristics, most notably the absence of negative feedback, but also the possible uncertainty in the positive feedback, as an observed user-item interaction may not necessarily indicate a preference of the user towards the item (see Section 2.1.2).

Matrix factorization for positive-only feedback

In one of the most representative works in this direction, Hu et al., (2008) proposed an adaptation of the rating-based MF model described previously to deal with positive-only feedback. As opposed to the rating-based SVD, which only considers the observed ratings, Hu et al.'s method models the full set of $|U| \cdot |I|$ interactions. Since negative feedback is not available in this scenario, the authors argue that the algorithm also has to model the missing information as an indirect source of negative user preferences. For such purpose, they introduce a parameter c_{ui} for each possible user-item pair that measures the confidence on the corresponding interaction, whether observed or not:

$$c_{ui} = 1 + \alpha k_{ui} \tag{2.22}$$

where k_{ui} is the count of implicitly collected interactions between user u and item i –such as number of clicks on a product web page in a e-commerce site, and the number of listening records of a given song in an online music provider–, and $\alpha > 0$ is a scaling parameter. When no interaction is observed, $k_{ui} = 0$ and the model assigns minimum confidence to the user-item pair, as it is unknown whether the lack of interaction is because the user really does not like the item, or just because the user does not know the item. Likewise, the more interactions are collected and the greater k_{ui}, the larger is the confidence on that observation. Moreover, focusing on the item ranking task, Hu et al.'s approach only aims to predict if the user will interact with the item, rather than the actual number of observations k_{ui}. Hence, a new set of variables is introduced so that $x_{ui} = 1$ if $k_{ui} > 0$, and $x_{ui} = 0$ otherwise.

Similarly to the SVD model for ratings, the recommendation score of item i for user u is estimated as the dot product of their corresponding latent feature vectors:

$$s(u, i) = \langle \vec{p}_u, \vec{q}_i \rangle \tag{2.23}$$

The model parameters $\vec{p}_u$ and $\vec{q}_i$ are again automatically learned by minimizing the mean squared error for the score predictions, but now accounting for the different confidence levels and the full set of possible user-item pairs:

$$\mathcal{L}(\mathbf{P}, \mathbf{Q}) = \sum_u \sum_i c_{ui} \left(x_{ui} - \langle \vec{p}_u, \vec{q}_i \rangle\right)^2 + \lambda \left(\|\mathbf{P}\|^2 + \|\mathbf{Q}\|^2\right) \tag{2.24}$$

Alternating Least Squares for positive-only feedback

Again, the loss function can be minimized with different numerical optimization techniques such as Stochastic Gradient Descent, but in (Hu et al., 2008) the authors propose an *Alternating Least Squares* (ALS) procedure that efficiently handles the greater cost of accounting for the missing values. Clearly, the loss function in Equation 2.24 involves many more terms than that of Equation 2.19, as the number of observed entries in the user-item matrix is usually very small due to the data sparsity.

The key observation behind ALS is that when one set of parameters is fixed, the optimization problem in Equation 2.24 is convex and analytically solvable using ordinary least-squares estimation. In particular, fixing the $\vec{q}_i$ parameters and setting the gradient with respect to $\vec{p}_u$ to zero yields the solution

$$\vec{p}_u = \left(\mathbf{Q}^\top \mathbf{C}^u \mathbf{Q} + \lambda \mathbf{I}\right)^{-1} \mathbf{Q}^\top \mathbf{C}^u \vec{x}_u \,. \tag{2.25}$$

where $\mathbf{I}$ is the $k \times k$ identity matrix, $\mathbf{C}^u$ is a $|I| \times |I|$ diagonal matrix with the c_{ui} values, and $\vec{x}_u$ is a column vector of length $|I|$ with the x_{ui} values. The same procedure can be applied by fixing the user factors, and optimizing the item factors, leading to the solution

$$\vec{q}_i = \left(\mathbf{P}^\top \mathbf{C}^i \mathbf{P} + \lambda \mathbf{I}\right)^{-1} \mathbf{P}^\top \mathbf{C}^i \vec{x}_i \,. \tag{2.26}$$

Similarly, $\mathbf{C}^i$ is a $|U| \times |U|$ diagonal matrix with the c_{ui} confidence values, and $\vec{x}_i$ is a column vector of length $|U|$ containing the binary values of x_{ui}.

Time complexity of Alternating Least Squares for positive-only feedback

As pointed out by the authors, the products $\mathbf{Q}^\top \mathbf{C}^u \mathbf{Q}$ and $\mathbf{P}^\top \mathbf{C}^i \mathbf{P}$ require time $\mathcal{O}(k^2|U|)$ and $\mathcal{O}(k^2|I|)$ for each user and item, respectively,

Algorithm 1 Alternating Least Squares training algorithm.

procedure ALS-TRAIN
  Initialize $\mathbf{P}, \mathbf{Q}$ at random
  repeat
    P step
      Fix $\mathbf{Q}$ and optimize all $\vec{p}_u$ in parallel using Equation 2.25
    Q step
      Fix $\mathbf{P}$ and optimize all $\vec{q}_i$ in parallel using Equation 2.26
  until convergence
end procedure

and represent a computational bottleneck during the training phase. However, these terms can be computed more efficiently noting that $\mathbf{Q}^\top\mathbf{C}^u\mathbf{Q} = \mathbf{Q}^\top\mathbf{Q} + \mathbf{Q}^\top(\mathbf{C}^u - \mathbf{I})\mathbf{Q}$, where $\mathbf{Q}^\top\mathbf{Q}$ is independent of the user and thus can be precomputed, and $\mathbf{C}^u - \mathbf{I}$ only has non-zero entries in the diagonal for the $|I(u)|$ items with $k_{ui} > 0$, which is much smaller than $|I|$. Considering the computation of the matrix inverse, the total complexity of Equation 2.25 for a single user is $\mathcal{O}(k^2|I(u)| + k^3)$. Likewise, the complexity for Equation 2.26 is $\mathcal{O}(k^2|U(i)| + k^3)$.

The main advantage of ALS is that the optimal factors for each user in Equation (2.25) can be computed in parallel once the item factors are fixed (P step). Symmetrically, once the user factors are obtained and fixed, the item factors in Equation 2.26 can be found for each item in parallel (Q step). This observation leads to the alternating nature of ALS, respectively fixing one set of parameters and optimizing the other until convergence is reached or for a given number of iterations, as illustrated in Algorithm 1.

Optimization of Alternating Least Squares for positive-only feedback

The ALS-based method by Hu et al., (2008) became the standard baseline for matrix factorization models with positive-only feedback, and has been extended and improved in subsequent works since it was first proposed. One notable paper by Pilászy et al., (2010) presents a new training procedure to boost the time complexity of the P step of each user to $\mathcal{O}(k^2 + k|I(u)|)$, and analogously the Q step. In order to achieve this significant improvement, the authors propose an approximate solution to the least-squares problem in each step. Rather than directly finding the k-dimensional solution as in Equations (2.25) and (2.26), which involves the costly computation of a matrix inverse, their approach iteratively solves k one-dimensional least squares problems, one for each latent dimension, much less expensive to solve. As reported in the paper, the loss of accuracy due to the approximate algorithm is small compared to the saved time for training. In subsequent work, Takács and Tikk, (2012) extended ALS to a ranking-based MF approach that learns to predict the relative ordering of items instead of individual point-wise scores. More recently, Paquet and Koenigstein, (2013) proposed a graph-based Bayesian model that is able to capture the meaning of missing values, distinguishing between a user disliking an item or being unaware of it.

2.4 EVALUATION OF RECOMMENDATION TECHNIQUES

In spite of two decades of study and developments, the evaluation of recommender systems (Gunawardana and Shani, 2015) is still an active research topic. As the focus of the community shifted from rating prediction to item ranking, distinct evaluation methodologies and metrics have been proposed to benchmark and compare the recommendation algorithms in the state of the art. In this section we review three issues concerning the evaluation of recommender systems, namely the setting or environment in which experiments are performed, the protocols used for preparing the testing data, and the metrics used to quantitatively compare the recommendation approaches.

2.4.1 *Experimental setting*

Offline experiments

Offline experiments represent the most popular approach to evaluate recommender systems in academic research, as they are relatively simple to design and cheap to conduct, which allows for faster benchmarking and comparison of recommendation approaches. In order to perform an offline experiment, first a (rating) dataset is collected as a source of user preferences in a particular domain, typically from an already deployed system. Some datasets have been made available for research purposes, such as the MovieLens[2] and LibraryThing[3] datasets. The approaches are then compared based on metrics that estimate the quality (e.g., in terms of precision, recall, coverage, and diversity) of their output on a common dataset, which is usually split into training and test subsets for building the recommendation models and evaluating the generated recommendations, respectively. Therefore, once the data is collected, the evaluation is independent from any external variables. This simplifies the experiments, but has the drawback that no other information about the users' experience with the system is available, such as their opinion about the usefulness and unexpectedness of the recommendations, their engagement with the system, and overall satisfaction.

Online experiments

Online experiments are suitable for systems already deployed with a sufficiently large base of users, commonly found in industry. Businesses evaluate their recommender systems analyzing the behavior of real customers using techniques such as A/B testing (Kohavi et al., 2009). A small fraction of the incoming traffic is redirected to an alternative system target of the evaluation, while the remaining users

2 MovieLens datasets, `http://grouplens.org/datasets/movielens`
3 LibraryThing datasets, `http://www.macle.nl/tud/LT`

keep interacting with the normal system, which acts as a baseline. The performance of the target system is measured relative to the baseline, typically in terms of click-through rate, dwell time, and number of visited links, or ultimately in terms of product revenue. This technique allows businesses to easily collect information directly from real users and test their recommendation approaches at little cost, but runs into the risk of degrading the experience of customers belonging to the test group. Moreover, uncontrolled external factors may have an impact on the collected data, misleading the results of the evaluation.

User studies

Finally, *user studies* (Knijnenburg and Willemsen, 2015) evaluate recommender systems by directly monitoring the interactions of a set of test users with the system in a controlled environment. The users' actions are recorded, and then used to assess the quality of the recommendations, both quantitatively based on metrics, and qualitatively asking the users about their experience with the system. For instance, Ekstrand et al., (2014) conducted a user study collecting information about the users' satisfaction and perception on the accuracy, novelty and diversity of recommendations generated by several baseline algorithms. User studies allow collecting more fine-grained information from the users than offline and online experiments, but are in contrast much more expensive to conduct, as they require recruiting users willing to participate in the experiments, usually with an economic reward. This in turn has the potential of attracting low performing, unrealistic users, who are only interested in the reward. Moreover, additional user recruitment is needed if a new system variant is developed or some information was not recorded during the original study.

2.4.2 *Evaluation methodologies*

Training, validation and test sets

The performance of recommendation algorithms is usually measured by comparing their item relevance predictions against a *ground truth* of items known to be relevant for the user. In order to perform a meaningful evaluation, the ground truth must be hidden to the recommendation algorithm, emulating the real-world case in which the system has to predict unknown, future user preferences. For this purpose, different protocols have been designed to partition the available (rating) data into a training set for building the recommendation models, and a testing set as the ground truth to evaluate the built models, i.e. $\mathcal{R} = \mathcal{R}_{\text{train}} \sqcup \mathcal{R}_{\text{test}}$. Sometimes a third set $\mathcal{R}_{\text{val}}$ is also held out for parameter tuning, selecting the model parameters that yield the best performance on this set. With the obtained parameters, an optimized model is then built with the training set, and next is evaluated on the test set, which is completely independent of the validation set in order to avoid biased evaluations of the recommendation performance $\mathcal{R} = \mathcal{R}_{\text{train}} \sqcup \mathcal{R}_{\text{val}} \sqcup \mathcal{R}_{\text{test}}$ (Hastie et al., 2009).

The majority of methodologies split the data at random, for instance selecting 80% of the user-item interactions for training, and the remaining 20% for test. Usually this procedure is repeated several times, averaging performance values in order to get a more robust evaluation. In this context, the more principled K-fold *cross-validation* methodology first splits the data into K roughly equal sets –the folds–, one of which is held out as ground truth and the remaining are used for training. The process is repeated K times, iteratively selecting each fold as test set (Sarwar et al., 2001). Instead of splitting the data randomly, another possibility is to perform a time-based split, when the timestamp of each interaction is available, and with the basis of using older interactions for training and recent ones for testing (Campos et al., 2014).

Cross-validation

A PROTOCOL FOR COLD START SITUATIONS We now describe a methodology specifically designed for evaluating the performance of recommender systems in user cold start situations, which was proposed by Kluver and Konstan, (2014), and which we shall follow in our experiments of Chapters 5 and 6. The dataset is split according to a user-based cross-validation protocol, i.e. the set of users is split in K sets. The data from the users in $K-1$ folds is used for training, and the data of the users in the remaining fold is used for testing. The user ratings (or the equivalently positive-only feedback records) of the test users are further split reserving m ratings for training, and the remaining for ground truth. In order to simulate cold start profiles of different sizes, the set of m training is iteratively downsampled discarding from 1 up to m ratings each time. This procedure avoids biases in metrics that are sensitive to the size of the test set, as the ground truth of each user remains constant for the different cold start levels.

Cross-validation for cold start users

2.4.3 *Evaluation metrics*

The area of recommender systems borrows a number of metrics from related fields, such as metrics of ranking quality from Information Retrieval, and metrics to evaluate both classification and regression models in Machine Learning. The choice of a metric has to be driven by the goal of the evaluation, as the metrics provide insight on distinct properties of the recommendations. Early papers focusing on the rating prediction task were most concerned with measuring the accuracy of their estimations. Hence, the performance was usually measured in terms of error-based metrics, such as the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE):

$$\mathrm{MAE} = \frac{1}{|\mathcal{R}_{\text{test}}|} \sum_{(u,i)\in\mathcal{R}_{\text{test}}} |r_{ui} - \hat{r}(u,i)| \tag{2.27}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathcal{R}_{\text{test}}} \sum_{(u,i)\in\mathcal{R}_{\text{test}}} (r_{ui} - \hat{r}(u,i))^2} \tag{2.28}$$

Instead of measuring error, ranking-based metrics analyze the ordering of the items in the list of recommendations. *Precision* at cutoff k, P@k, computes the fraction of items in the top k of the ranking that are relevant for the user, and *Recall* at k, R@k, measures the fraction of relevant items for the user that were recommended in the top k positions. Global scores for these metrics are obtained by averaging their values for all the test users (Baeza-Yates and Ribeiro-Neto, 1999):

$$P@k = \frac{1}{|\mathcal{U}|} \sum_{u\in\mathcal{U}} \frac{|Rel_u@k|}{k} \tag{2.29}$$

$$R@k = \frac{1}{|\mathcal{U}|} \sum_{u\in\mathcal{U}} \frac{|Rel_u@k|}{|Rel_u|} \tag{2.30}$$

where Rel_u is the set of relevant items for user u, and $Rel_u@k$ is the set of relevant items of user u that are in the top k positions of the recommendation ranking.

The Mean Average Precision (MAP) metric considers the relative order of the relevant items in the recommendation ranking by computing the precision score after each one of them is found:

$$AP(u) = \frac{1}{|Rel_u|} \sum_{n=1}^{k} P@n \cdot \mathbb{1}(i_n \in Rel_u)$$

$$\mathrm{MAP} = \frac{1}{|\mathcal{U}|} \sum_{u\in\mathcal{U}} AP(u) \tag{2.31}$$

where the precision P@n is computed as in Equation 2.29, and i_n is the item in the n-th position of the recommendation list of user u.

The normalized Discounted Cumulative Gain (nDCG) proposed by Järvelin and Kekäläinen, (2002) is a suitable metric when there are multiple levels of relevance in the ground truth. The more relevant an item, the more it contributes to the quality if it is recommended, but adjusted to its relative position in the ranking:

$$DCG_u@k = \sum_{n=1}^{k} \frac{2^{rel_u(i_n)} - 1}{\log(n+1)}$$

$$nDCG@k = \frac{1}{|\mathcal{U}|} \sum_{u\in\mathcal{U}} \frac{DCG_u@k}{IDCG_u@k} \tag{2.32}$$

$\mathrm{rel}_u(i_n)$ is the graded relevance for user u of the item in the n-th position of the ranking, and $\mathrm{IDCG}_u@k$ is the discounted cumulative gain of the ideal ranking for user u at cutoff k.

As an alternative, the Mean Reciprocal Rank (MRR) measures the inverse of the position in the ranking of the first relevant recommended item (Baeza-Yates and Ribeiro-Neto, 1999):

$$\mathrm{MRR} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{\mathrm{rank}_u} \tag{2.33}$$

where rank_u is the position of the first relevant item recommended for user u. If no relevant item is recommended for u, then $\mathrm{rank}_u \triangleq \infty$.

The previous metrics estimate the performance of a recommender system in terms of ranking quality for the users, but some metrics focus on evaluating the system from the point of view of the business. For instance, the *user coverage* is defined as the ratio of users for whom the system is able to provide recommendations. Likewise, the *item coverage* measures the proportion of items in the system catalog that are recommended to the users. Furthermore, Shannon's entropy of the item distribution across the recommendation lists is useful to see whether the all items are uniformly recommended or if some are suggested more often (Gunawardana and Shani, 2015):

$$\mathcal{H} = -\sum_{i \in \mathcal{I}} p(i) \log p(i) \tag{2.34}$$

where $p(i)$ is the fraction of users that get item i in their recommendation list.

Traditionally, recommender systems were mostly evaluated in terms of the accuracy of their rating predictions or of their rankings, using metrics like the ones mentioned above. Nonetheless, it has been argued that accuracy is only one dimension of the quality of recommendations (McNee et al., 2006), and during the last decade new metrics have been proposed to address other aspects such as *novelty* and *diversity* (Vargas and Castells, 2011).

Novelty metrics aim to measure to what extent recommended items are unknown for the user, based on the items preferred by the user in the past (Gunawardana and Shani, 2015), and can be understood as the different between the user's present and past experience (Castells et al., 2015). The novelty of the recommendations can be assessed in user studies by asking the users whether they are familiar with the recommended items (Celma and Herrera, 2008; Ekstrand et al., 2014). For a comprehensive review of novelty metrics for offline experiments, the reader is referred to (Castells et al., 2015). Diversity metrics, on the other hand, aim to measure how different the recommended items are with respect to each other. Intra-List Diversity (ILD) measures diversity as the average pairwise distance between the items in the recommendation list R_u (Smyth and McClave, 2001; Ziegler et al., 2005):

$$\mathrm{ILD} = \frac{1}{|R_u|(|R_u|-1)} \sum_{i,j \in R_u} \mathtt{dist}(i,j) \tag{2.35}$$

where the function $\mathtt{dist}(i,j)$ measures the distance between two items, typically based on content features. Later, Vargas et al., (2014) introduced the Binomial Diversity framework, measuring genre-based diversity in terms of coverage and redundancy in the recommendation list:

$$\mathit{BinCoverage} = \prod_{g \notin G_R} p(X_g = 0)^{1/|G_R|} \tag{2.36}$$

$$\mathit{BinRedundancy} = \prod_{g \in G_R} p(X_g \geqslant k_g | X_g > 0)^{1/|G_R|} \tag{2.37}$$

$$\mathit{BinDiversity} = \mathit{BinCoverage} \cdot \mathit{BinRedundancy} \tag{2.38}$$

G_R is the set of genres from the items in the recommendation list R_u, and X_g is a random variable following a binomial distribution of $|R_u|$ trials, counting the number of items k_g in the list that contain genre g.

Other metrics have been considered in the literature, which we do not discuss here, but manifest desirable aspects for a recommender system, such as confidence and trust, serendipity, robustness, privacy, and scalability (Gunawardana and Shani, 2015).

3 CROSS-DOMAIN RECOMMENDER SYSTEMS

Recommender systems have been successfully used in numerous domains and applications to identify potentially relevant items for users according to their preferences. Even though the majority of recommender systems focus on a single domain or type of item, there are cases in which providing the user with *cross-domain recommendations* could be beneficial. For instance, in large e-commerce sites users express feedback for items of different types, and in social networks users often share their tastes and interests on a variety of topics. In these cases, rather than exploiting user preference data from each domain independently, recommender systems could exploit more exhaustive, multi-domain user models that allow generating item recommendations spanning several domains. Furthermore, utilizing additional knowledge from related, auxiliary domains could help improve the quality of item recommendations in a target domain, e.g., addressing the data sparsity and cold start problems.

In this chapter we introduce cross-domain recommender systems, providing a unified categorization and analysis of the literature on the topic in different research areas. Specifically, in Section 3.1 we introduce and motivate the consideration of cross-domain recommendations, and in Section 3.2 we formulate the cross-domain recommendation problem, describing its main tasks and goals. Next, in Section 3.3 we provide a general categorization of cross-domain recommendation techniques. In Section 3.4 and Section 3.5 we review existing cross-domain recommendation approaches, distinguishing knowledge aggregation and knowledge linkage/transfer approaches. Finally, in Section 3.6 we overview issues on the evaluation of cross-domain recommendations.

3.1 MOTIVATIONS FOR CROSS-DOMAIN RECOMMENDATIONS

Nowadays, the majority of recommender systems offer recommendations for items belonging to a single domain. For instance, Netflix[1] recommends movies and TV shows, Spotify[2] recommends songs and music albums, and Barnes & Noble[3] recommends books. These domain-specific systems have been successfully deployed by numerous web platforms, and the single-domain recommendation function-

1 Netflix streaming media and video provider, `https://www.netflix.com`
2 Spotify digital music service, `https://www.spotify.com`
3 Barnes & Noble online bookseller, `http://www.barnesandnoble.com`

ality is not perceived as a limitation, but rather pitched as a focus on a certain market.

Nonetheless, in large e-commerce sites such as Amazon.com[4] and eBay[5] users often provide feedback for items from multiple domains, and in social networks like Facebook[6] and Twitter[7] users express their tastes and interests for a variety of topics. It may, therefore, be beneficial to leverage all the available user data provided in various systems and domains, in order to generate more encompassing user models and better recommendations. Instead of treating each domain (e.g., movies, music and books) independently, knowledge acquired in a *source* domain could be transferred to and exploited in another *target* domain. The research challenge of transferring knowledge, and the business potential of delivering recommendations spanning across multiple domains, have triggered an increasing interest in *cross-domain recommendations*.

Examples of use cases motivating cross-domain recommendations

Consider two motivating use cases for cross-domain recommendations. The first refers to the *cold start* problem, which hinders the recommendation generation due to the lack of sufficient information about users or items. In a cross-domain setting, a recommender may draw on information acquired from other domains to alleviate such problem, e.g. a user's favorite movie genres may be derived from her favorite book genres. The second refers to the generation of personalized *cross-selling* or bundle recommendations for items from multiple domains, e.g., a movie accompanied by a music album similar to the soundtrack of the movie. This recommendation may be informed by user's movie tastes, but may not be extracted from rating correlations within a joined movie-music rating matrix.

User and item inter-domain dependencies

These use cases are underpinned by an intuitive assumption that there are correspondences between user and item profiles in the source and target domains. This assumption has been validated in several marketing, behavioral, and data mining studies, which uncover strong dependencies between different domains (Shapira et al., 2013; Winoto and Tang, 2008). Cross-domain recommender systems leverage these dependencies through considering, for example, overlaps between the user or item sets, correlations between user preferences, and similarities of item attributes. Then, they apply a variety of techniques for enriching the knowledge of the target domain, and improving the quality of recommendations generated therein.

Cross-domain recommendation is a challenging and still largely under explored topic. Although it has been studied from several angles, an agreed upon definition of the problem has not emerged yet, and no work has analyzed and classified the existing approaches. In this chapter we provide a unified view and formulation of the cross-

4 Amazon electronic commerce site, https://www.amazon.com
5 eBay consumer-to-consumer and business-to-consumer sales, http://www.ebay.com
6 Facebook social network, https://www.facebook.com
7 Twitter online news and social networking service, https://twitter.com

domain recommendation problem, survey the state of the art in cross-domain recommender systems, categorize the methods for establishing and exploiting links between diverse domains, and compare outcomes of prior work.

3.2 THE CROSS-DOMAIN RECOMMENDATION PROBLEM

The cross-domain recommendation problem has been addressed from various perspectives in different research areas. It has been handled by means of user preference aggregation and mediation strategies for cross-system personalization in User Modeling (Abel et al., 2013; Berkovsky et al., 2008; Shapira et al., 2013), as a potential solution to mitigate the cold start and sparsity problems in Recommender Systems (Cremonesi et al., 2011b; Shi et al., 2011; Tiroshi et al., 2013), and as a practical application of knowledge transfer in Machine Learning (Gao et al., 2013; Li et al., 2009a; Pan et al., 2010).

Aiming to unify perspectives, we provide a generic formulation of the cross-domain recommendation problem, focusing on existing domain notions (Section 3.2.1) and cross-domain recommendation tasks and goals (Section 3.2.2), and discuss the possible scenarios of data overlap between domains (Section 3.2.3).

3.2.1 *Definitions of domain*

Notions of domain

In the literature researchers have considered distinct notions of domain. For instance, some have treated items like *movies* and *books* as belonging to different domains, while others have considered items such as *action movies* and *comedy movies* as different domains. To the best of our knowledge, in the context of recommender systems research, there have been no attempts to define the concept of *domain*. Here we distinguish several domain notions according to the attributes and types of recommended items. Specifically, we consider that *domain* may be defined at four levels (see illustration in Figure 3.1):

- *(Item) Attribute level*. Recommended items are of the same type, having the same attributes. Two items are considered as belonging to distinct domains if they differ in the value of certain attribute. For instance, two movies belong to distinct domains if they have different genres, like action and comedy movies. This definition of domain is rather borderline, and is mainly used as a way to increase the diversity of recommendations (e.g., we may wish to recommend some thriller movies even to users who only watch comedy movies).

- *(Item) Type level*. Recommended items are of similar types and share some attributes. Two items are considered as belonging to

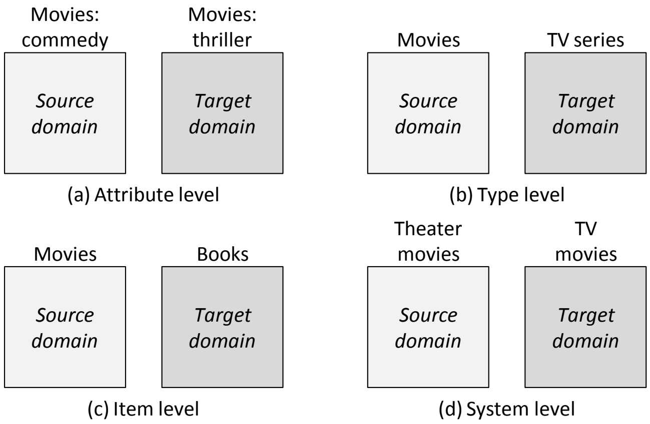

Figure 3.1: Notions of domain according to attributes and types of recommended items. (a) *Attribute level*: same type of items (movies) with different values of certain attribute (genre). (b) *Type level*: similar types of items (movies and TV shows), sharing some of their attributes. (c) *Item level*: different types of items (books and movies). (d) *System level*: same type of items (movies) on different systems (theater and TV).

distinct domains if they have different attribute subsets. For instance, movies and TV shows belong to distinct domains, since although they have several attributes in common (title, genre), they still differ with respect to some others (e.g., the live attribute for TV shows).

- *Item level*. Recommended items are not of the same type, differing in most, if not all, of their attributes. For instance, movies and books belong to different domains, even though they have some attributes in common (title, release/publication year).

- *System level*. Recommended items belong to distinct systems, which are considered as different domains. For instance, movies rated in the MovieLens recommender, and movies watched in the Netflix video streaming service.

Notions of domain and types of user preferences in the literature

In Table 3.1 we summarize the considered notions of domains, addressed domains, and used datasets/systems in a significant number of prior works on cross-domain user modeling and recommendation. It can be seen that the majority of the papers considers domains at the item (about 55%) and system (24%) levels. The most frequently addressed domains are movies (75%), books (57%), music (39%) and TV (18%). In this context, we note that around 10% of the papers addresses various domains, by exploiting user preference data from multi-domain systems like Amazon and Facebook. Analyzing the pairs of domains frequently addressed, we observe that movies are often crossed with books (33%), music (19%), and TV (7%), whereas books are crossed with music (14%) and TV (10%).

Table 3.1: Summary of domain notions, domains, and user preference datasets used in the cross-domain user modeling and recommendation literature.

Domain notion	Domains	User preferences - *datasets*	References
item attribute	book categories	ratings - *BookCrossing*	Cao et al., 2010
	movie genres	ratings - *EachMovie*	Berkovsky et al., 2007b
		ratings - *MovieLens*	Cao et al., 2010; Lee and Seung, 2001
item type	books, movies, music	ratings - *Amazon*	Hu et al., 2013; Loni et al., 2014
	books, games, music, movies & TV shows	ratings	Winoto and Tang, 2008
item	books, movies	ratings - *BookCrossing, MovieLens/EachMovie*	Gao et al., 2013; Li et al., 2009a,b
		ratings, tags - *LibraryThing, MovieLens*	Enrich et al., 2013; Shi et al., 2011; Zhang et al., 2010
		ratings, transactions	Azak, 2010
		ratings - *Imhonet*	Sahebi and Brusilovsky, 2013
		ratings - *Douban*	Zhao et al., 2013
	movies, music	thumbs up - *Facebook*	Shapira et al., 2013
	books, movies, music	tags - *MovieLens, Last.fm, LibraryThing*	Fernández-Tobías et al., 2013
	books, movies, music, TV shows	thumbs up - *Facebook*	Cantador et al., 2013; Tiroshi and Kuflik, 2012; Tiroshi et al., 2013
	music, tourism	semantic concepts	Fernández-Tobías et al., 2011; Kaminskas et al., 2013
	restaurants, tourism	ratings, transactions	Chung et al., 2007
	various domains	tags - *Delicious, Flickr*	Szomszor et al., 2008a,b
system	movies	ratings - *Netflix*	Cremonesi et al., 2011b; Zhao et al., 2013
		ratings - *Douban, Netflix*	Zhao et al., 2013
		ratings - *MovieLens, Moviepilot, Netflix*	Pan and Yang, 2013; Pan et al., 2012
	music	tags - *Delicious, Last.fm*	Loizou, 2009
		tags - *Blogger, Last.fm*	Stewart et al., 2009
	various domains	tags - *Delicious, Flickr, StumbleUpon, Twitter*	Abel et al., 2011, 2013
		click-through data - *Yahoo! services*	Low et al., 2011

The table also shows the utilized types of user preferences: ratings (61%), tags (29%), thumbs up (14%), transaction history (7%), and click-through data (4%). Although only a few papers use semantic concepts as user preferences, in some papers social tags are transformed into concepts from WordNet or Wikipedia. Overall, about 14% of the papers use semantic-based user preferences.

3.2.2 *Cross-domain recommendation tasks and goals*

Source and target domains

The research on cross-domain recommendation generally aims to exploit knowledge from a source domain $\mathcal{D}_S$ to perform or improve recommendations in a target domain $\mathcal{D}_T$. Analyzing the literature, we observe that the addressed tasks are diverse, and a consensual definition of the cross-domain recommendation problem has not been formulated yet. Hence, some researchers have proposed models aimed to provide jointly diverse recommendations of items belonging to multiple domains, whereas others have developed methods to alleviate cold start and sparsity situations in a target domain by using information from source domains.

Cross-domain recommendation tasks

Aiming to provide a unified formulation of the cross-domain recommendation problem, we first identify the tasks proposed when providing recommendations across domains. Without loss of generality, we consider two domains $\mathcal{D}_A$ and $\mathcal{D}_B$ (the definitions are extensible to more than two domains). Let $\mathcal{I}_A$ and $\mathcal{I}_B$ be their sets of items. The items do not necessarily have preferences from users of the domain, but may have content-based attributes that establish their membership to the domain.

We distinguish the following three recommendation tasks:

- *Multi-domain recommendation*: recommend together items in both domains, i.e., recommend groups of items in $\mathcal{I}_A \cup \mathcal{I}_B$.
- *Cross-selling*: recommend items in a new domain, different to that where the users expressed their preferences, i.e., recommend items in $\mathcal{I}_B$ to users with preferences for items in $\mathcal{I}_A$.
- *Linked domains exploitation*: leverage preferences from other domains to improve recommendations in a target domain, i.e., recommend items in $\mathcal{I}_B$ by exploiting knowledge relating $\mathcal{D}_A$ and $\mathcal{D}_B$.

Multi-domain recommendations

Multi-domain approaches have mainly focused on the provision of cross-system personalization, by jointly considering user preferences for items in various systems. To perform this type of recommendations, a significant overlap between user preferences in distinct domains is needed. This is becoming more and more feasible, since nowadays users maintain profiles in various social media, and there are interconnecting mechanisms for both cross-system interoperability (Carmagnola et al., 2011) and cross-system user identification (Carmagnola and Cena, 2009). Moreover, multi-domain recommendation is tightly connected to bundle recommendation, in which a system aims to jointly suggest items from different domains and provide a better value to the users when recommended together, e.g., personalized car renting, restaurant, and touristic attraction visit offers together with hotel book recommendations.

Cross-selling recommendations

Cross-selling approaches have been mainly proposed to provide recommendations in e-commerce sites, where they can increase customer satisfaction (Driskill and Riedl, 1999; Kitts et al., 2000), and consequently, their loyalty and the businesses profitability. For such purposes, in general, these approaches aim to exploit knowledge-based links between items. Cross sells work well with items that require accessories, e.g., in a technology e-commerce site, a user who just purchased a laptop may be interested in certain mouse and extra battery for that computer, even if the user had not shown any preference for such types of items.

Linked domains exploitation

Linked domains exploitation approaches have been mainly explored to improve the recommendations in a target domain where there is a scarcity of user preferences, either at the user level (the cold start problem) or at the community level (the data sparsity problem). To deal with these situations, a common solution is to enrich or enhance the available knowledge in the target domain with knowledge from related domains. Hence, to perform this type of recommendations, some data relations or overlaps between domains are needed, and approaches aim to establish explicit or implicit knowledge-based links between the domains, e.g., in an online bookseller, suggesting certain types of novels to users who expressed preferences for certain movie genres in an external online social network.

For the sake of simplicity, we consider the three recommendation tasks together, as a single formulation of the cross-domain recommendation problem, although in Section 3.4 and Section 3.5 we review specific approaches for each task.

From both the research and practical perspectives, it is important to match the recommendation algorithms to the task in hand. For this reason, we initially present a taxonomy of cross-domain recommendation goals. The taxonomy is described in a solution-agnostic way: each problem is defined based solely on its goals —without discussing how they are solved, which will be done in Section 3.3.

Cross-domain recommendation goals

At the first level of the taxonomy, we consider the three recommendation tasks presented in Section 3.2.2, namely *multi-domain*, *linked-domain*, and *cross-selling* tasks. At the second level, we distinguish the specific goals addressed by cross-domain recommenders, which are the rows of Table 3.2. We identify the following goals:

- *Addressing the system cold start problem (system bootstrapping*. This is related to situations in which a recommender is unable to generate recommendations due to an initial lack of user preferences. One possible solution is to bootstrap the system with preferences from another source outside the target domain.

- *Addressing the new user problem*. When a user starts using a recommender, this has no knowledge of the user's tastes and interests, and cannot produce personalized recommendations. This may be solved by exploiting the user's preferences collected in a different source domain.
- *Addressing the new item problem (cross-selling of products)*. When a new item is added to a catalog, it has no prior ratings, so it will not be recommended by a collaborative filtering system. This problem is particularly evident when cross-selling new products from different domains.
- *Improving accuracy (by reducing sparsity)*. In many domains, the average number of ratings per user and item is low, which may negatively affect the quality of the recommendations. Data collected outside the target domain can increase the rating density, and thus may upgrade the recommendation quality.
- *Improving diversity*. Having similar, redundant items in a recommendation list may not contribute much to the user's satisfaction. The diversity of recommendations can be improved by considering multiple domains, as this may provide a better coverage of the range of user preferences.
- *Enhancing user models*. The main goal of cross-domain user modeling applications is to enhance user models. Achieving this goal may have personalization-oriented benefits such as (i) discovering new user preferences for the target domain (Stewart et al., 2009; Szomszor et al., 2008a), (ii) enhancing similarities between users and items (Abel et al., 2011; Berkovsky et al., 2008), and (iii) measuring vulnerability in social networks (Goga et al., 2013; Jain et al., 2013).

Table 3.2 shows the mapping between the above recommendation goals.

3.2.3 *Cross-domain recommendation scenarios*

Inter-domain relations

As discussed in (Fernández-Tobías et al., 2012), in the context of a cross-domain recommendation task, domains can be explicitly or implicitly linked by means of content-based (CB) or collaborative filtering (CF) characteristics associated with users and/or items, such as ratings, social tags, semantic relations, and latent factors.

Let $\mathcal{X}^{\mathcal{U}} = \{x_1^{\mathcal{U}}, \cdots, x_m^{\mathcal{U}}\}$ and $\mathcal{X}^{\mathcal{I}} = \{x_1^{\mathcal{I}}, \cdots, x_n^{\mathcal{I}}\}$ be the sets of characteristics utilized to represent the users and items, respectively. Two domains $\mathcal{D}_S$ and $\mathcal{D}_T$ are linked if $\mathcal{X}_S^{\mathcal{U}} \cap \mathcal{X}_T^{\mathcal{U}} \neq \varnothing$ or $\mathcal{X}_S^{\mathcal{I}} \cap \mathcal{X}_T^{\mathcal{I}} \neq \varnothing$,

Table 3.2: Summary of cross-domain recommendation approaches based on goals.

Goal	References
Cold start	Shapira et al., 2013
New user	Berkovsky et al., 2007a,b, 2008; Braunhofer et al., 2013; Cremonesi et al., 2011b; Hu et al., 2013; Low et al., 2011; Nakatsuji et al., 2010; Sahebi and Brusilovsky, 2013; Tiroshi and Kuflik, 2012; Winoto and Tang, 2008
New item	Kaminskas et al., 2013
Accuracy	Cao et al., 2010; Gao et al., 2013; Li et al., 2009a,b, 2011; Loni et al., 2014; Moreno et al., 2012; Pan et al., 2008; Pan and Yang, 2013; Pan et al., 2010, 2012; Shi et al., 2011; Stewart et al., 2009; Tang et al., 2011; Tiroshi et al., 2013; Zhang et al., 2013; Zhang et al., 2010; Zhao et al., 2013
Diversity	Winoto and Tang, 2008
User model	Abel et al., 2011, 2013; Fernández-Tobías et al., 2013; Goga et al., 2013; Jain et al., 2013; Szomszor et al., 2008b

i.e., if they share user or item characteristics. In a realistic setting, due to the heterogeneity of domain representations, one may need to set functions that map characteristics between domains, i.e., $f : \mathcal{X}_S^{\mathcal{U}} \rightarrow \mathcal{X}_T^{\mathcal{U}}$ and $g : \mathcal{X}_S^{\mathcal{I}} \rightarrow \mathcal{X}_T^{\mathcal{I}}$. For instance, to link movies and books, a mapping function could identify users registered in two systems, $f(u_{i,movie\,system}) = u_{j,book\,system}$, or could link related genres, $g(comedy_{movie\,system}) = humor_{book\,system}$.

Next, we describe representative examples of user and item characteristics, as well as their inter-domain relations and data overlap scenarios.

- *Content-based relations between domains*. In CB systems, a set of content or metadata features $\mathcal{F} = \{F_1, \cdots, F_n\}$ – e.g., keywords, properties, and categories – describes both user preferences and item attributes, i.e., $\mathcal{X}^{\mathcal{U}} \subseteq \mathcal{F}, \mathcal{X}^{\mathcal{I}} \subseteq \mathcal{F}$. In general, a user profile is composed of a vector, where each component reflects the degree to which the user likes or is interested in a specific feature, and similarly, an item profile is composed of a vector whose components reflect the relevance of the features to the item. An overlap between domains $\mathcal{D}_S$ and $\mathcal{D}_T$ occurs when $\mathcal{X}_S^{\mathcal{U}} \cap \mathcal{X}_T^{\mathcal{U}} \neq \varnothing$ and $\mathcal{F}_S \cap \mathcal{F}_T \neq \varnothing$.

- *Collaborative filtering-based relations between domains*. In CF systems, user preferences are modeled as a matrix $\mathbf{R} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$, in which an element R_{ui} is the rating assigned by user u to item i. Thus, $\mathcal{X}^{\mathcal{U}} = \mathcal{I}$ ($\mathcal{I}$ being the rated items), and domains $\mathcal{D}_S$ and $\mathcal{D}_T$ overlap when $\mathcal{X}_S^{\mathcal{U}} \cap \mathcal{X}_T^{\mathcal{U}} \neq \varnothing$, i.e., $\mathcal{I}_S \cap \mathcal{I}_T \neq \varnothing$. An equivalent reasoning can be done for items, to derive that $\mathcal{X}^{\mathcal{I}} = \mathcal{U}$ ($\mathcal{U}$ be-

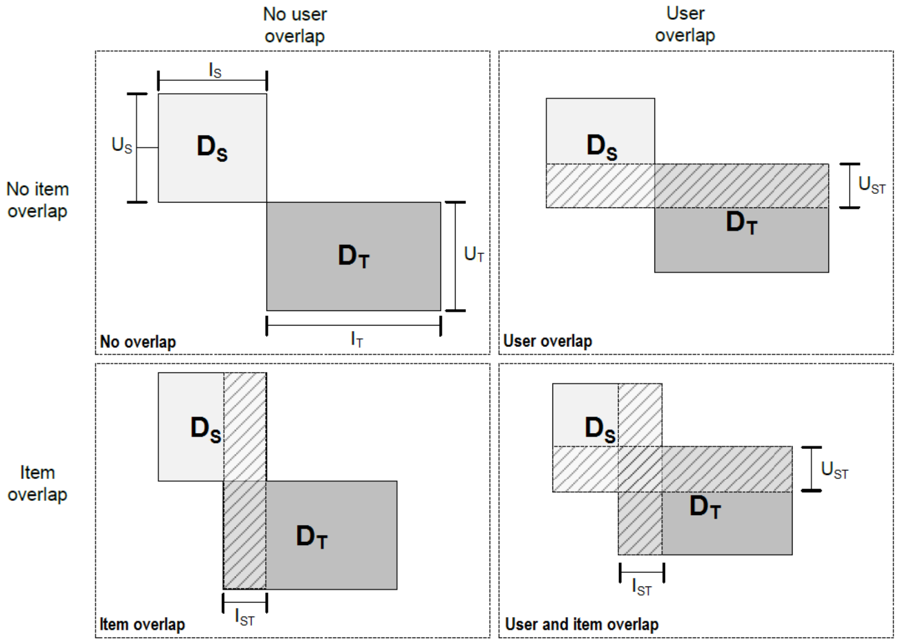

Figure 3.2: Scenarios of data overlap between user and item sets in two domains $\mathcal{D}_S$ and $\mathcal{D}_T$: *no overlap*, *user overlap*, *item overlap*, and *user and item overlap*.

ing the users with ratings), and that $\mathcal{D}_S$ and $\mathcal{D}_T$ overlap when $\mathcal{X}_S^{\mathcal{I}} \cap \mathcal{X}_T^{\mathcal{I}} \neq \varnothing$, i.e., $\mathcal{U}_S \cap \mathcal{U}_T \neq \varnothing$.

Moreover, as explained in subsequent sections, approaches have been proposed to represent users and/or items in lower dimension spaces, called *latent factors*, in which the above vector representations are valid. In these cases, if $\mathbf{U}$ and $\mathbf{I}$ denote the sets of user and item latent factors, respectively, then $\mathcal{X}^{\mathcal{U}} = \mathbf{U}$ and $\mathcal{X}^{\mathcal{I}} = \mathbf{I}$.

Data overlap between domains

As shown in Figure 3.2, for the above types of relations, and generalizing the possible cross-domain CF cases identified by Cremonesi et al., (2011b), four scenarios of data overlap between two domains $\mathcal{D}_S$ and $\mathcal{D}_T$ can exist:

- *No overlap*. There is no overlap between users and items in the domains, i.e., $\mathcal{U}_{ST} = \mathcal{U}_S \cap \mathcal{U}_T = \varnothing$ and $\mathcal{I}_{ST} = \mathcal{I}_S \cap \mathcal{I}_T = \varnothing$.

- *User overlap*. There are some common users who have preferences for items in both domains, i.e., $\mathcal{U}_{ST} \neq \varnothing$, but every item belongs to a single domain. This is the case, for instance, where some users rated movies and books.

- *Item overlap*. There are some common items that have been rated by users from both domains, i.e., $\mathcal{I}_{ST} \neq \varnothing$. This is the case, for instance, where two IPTV providers share a catalog of TV programs, which may be rated in each system.

- *User and item overlap*. There is overlap between both the users and items, i.e., $\mathcal{U}_{ST} \neq \varnothing$ and $\mathcal{I}_{ST} \neq \varnothing$.

3.3 CATEGORIZATION OF CROSS-DOMAIN RECOMMENDATION METHODS

As discussed in Section 3.2, cross-domain recommendation has been addressed from various perspectives in distinct research areas. This has entailed the development of a wide array of recommendation approaches, which in many cases are difficult to compare due to the user preferences they use, the cross-domain scenario they deal with, and the algorithms and data on which they are based. Moreover, published reviews of the research literature and categorizations of existing approaches (Cremonesi et al., 2011b; Fernández-Tobías et al., 2012; Li, 2011; Pan and Yang, 2010) have not reflected the entire complexity of the space. In this section, we categorize and propose a unifying schema for the existing cross-domain recommendation techniques.

Seminal work on cross-domain recommendation

Chung et al., (2007) presented in their seminal research a framework that provides integrated recommendations for items that may be of different types, and may belong to different domains. The framework accounts for three levels of recommendation integration: *single item type recommendations*, which consist of items of the same type, *cross item type recommendations*, which consist of items of different types that belong to the same domain, and *cross domain recommendations*, which consist of items whose types belong to different domains. The authors stated that integrated recommendations can be generated by following at least three approaches:

- General filtering: instantiating a recommendation model for multiple item types that may belong to different domains.
- Community filtering: using ratings shared among several communities or systems that may deal with different item types and domains.
- Market basket analysis: applying data mining to extrapolate hidden relations between items of different types/domains and to build a model for item filtering.

In (Loizou, 2009), Loizou identified three main trends in cross-domain recommendation research. The first focuses on compiling unified user profiles appropriate for cross-domain recommendations (González et al., 2005). This is considered as an integration of domain-specific user models into a single, unified multiple-domain user model, which is subsequently used to generate recommendations. The second involves profiling user preferences through monitoring their interactions in individual domains (Kook, 2005), which can be materialized through agents that learn single-domain user preferences and gather them from multiple domains to generate recommendations. The third deals with combining (or mediating) information from several single-domain recommender systems (Berkovsky et al., 2007a). A number of strategies for mediating single-domain CF systems were considered: exchange of ratings, exchange of user neighborhoods, exchange of user similarities, and exchange of recommendations.

Categorizing cross-domain collaborative filtering approaches

Based on these trends, Cremonesi et al., (2011b) surveyed and categorized cross-domain CF systems. They enhanced Loizou's categorization by considering a more specific grouping of approaches:

- Extracting association rules from rating behavior in a source domain, and using extracted rules to suggest items in a target domain, as proposed by Lee and Seung, (2001).
- Learning inter-domain rating-based similarity and correlation matrices, as proposed by Cao et al., (2010) and Zhang et al., (2010).
- Merging estimations of rating probability distributions in source domains to generate recommendations in a target domain, as proposed by Zhuang et al., (2010).
- Transferring knowledge between domains to address the rating sparsity problem in a target domain, as proposed by Li et al., (2009a,b) and Pan et al., (2010, 2011).

Categorizing transfer learning techniques

For the last group, Li, (2011) presented a survey of transfer learning techniques in cross-domain CF. There, Li proposed an alternative categorization based on types of domain. Specifically, the author distinguished (i) *system domains* that are associated with different recommenders, and represent a scenario where the data in a target recommender are very sparse, while the data in related recommenders are abundant; (ii) *data domains* that are associated with multiple sources of heterogeneous data, and represent a scenario where user data in source domains (e.g., binary ratings) can be obtained easier than in a target domain (e.g., five-star ratings); and (iii) *temporal domains* that are associated with distinct data periods, and represent a scenario where temporal user preference dynamics can be captured. For these categories, Li considered three recommendation strategies differing in the knowledge transferred between domains:

- Rating pattern sharing, which aims to factorize single-domain rating matrices utilizing user/item groups, encode group-level rating patterns, and transfer knowledge through the encoded patterns (Li et al., 2009a,b, 2011).
- Rating latent feature sharing, which aims to factorize single-domain rating matrices using latent features, share latent feature spaces across domains, and transfer knowledge between domains through the latent feature matrices (Pan and Yang, 2013; Pan et al., 2010, 2011, 2012).

- Domain correlating, which aims to factorize single-domain rating matrices using latent features, explore correlations between latent features in single domains, and transfer knowledge between domains through such correlations (Cao et al., 2010; Shi et al., 2011; Zhang et al., 2010).

Pan and Yang, (2010) identified in a survey of transfer learning for machine learning applications three main questions to be faced: (i) *what* to transfer —which knowledge should be transferred between domains; (ii) *how* to transfer –which learning algorithms should be exploited to transfer the discovered knowledge; and (iii) *when* to transfer –in which situations the knowledge transfer knowledge is beneficial. Focusing on the what and how questions, Pan et al. proposed in *Pan2011* and (Pan et al., 2010) a two-dimensional categorization of transfer learning-based approaches for cross-domain CF. The first dimension takes the type of transferred knowledge into account, e.g., latent rating features, encoded rating patterns, and rating-based correlations and covariances. The second dimension considers the algorithm, and distinguishes between adaptive and collective approaches, assuming, respectively, the existence of rating data only in the source domain, and in both the source and target domains.

Two-dimension categorization of cross-domain recommendation approaches

In a more recent survey, Fernández-Tobías et al., (2012) went beyond CF recommendations, taking into account approaches that establish cross-domain relationships not necessarily based on ratings. They identified three directions to address the cross-domain recommendation problem. The first is through the integration of single-domain user preferences into a *unified user model*, which implies aggregating user profiles from multiple domains ("compile unified profiles" in (Loizou, 2009)), and the mediation of user models across domains ("profile through monitoring" in (Loizou, 2009)). The second direction aims to *transfer knowledge* from a source domain to a target domain, and includes approaches that exploit recommendations generated for a source domain in a target domain ("mediating information" in (Loizou, 2009)), and approaches based on transfer learning, surveyed in (Li, 2011). The third direction is about *establishing explicit relations* between domains, which may be based either on content-based relations between items or on rating-based relations between users/items. The authors then proposed a two-dimensional categorization of cross-domain recommendation approaches: (i) according to the type of relations between domains: *content-based relations* (item attributes, tags, semantic properties, and feature correlations) vs. *rating-based relations* (rating patterns, rating latent factors, and rating correlations); and (ii) according to the goal of the recommendation task: *adaptive models*, which exploit knowledge from a source domain to generate recommendations in a target domain, vs. *collective models*, which are built using data from several domains to improve recommendations in a target domain.

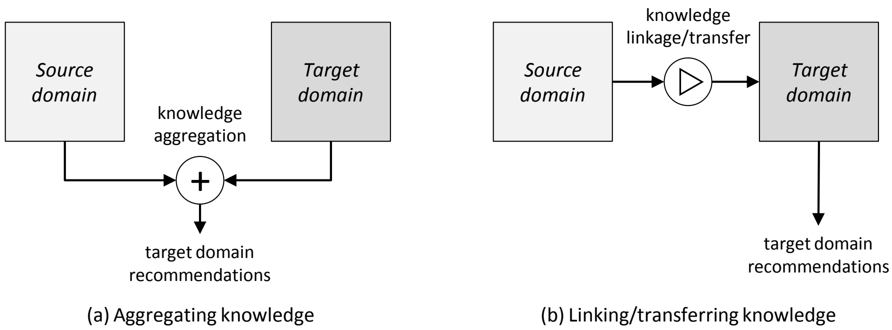

Figure 3.3: Exploitation of knowledge in cross-domain recommendation.

Proposed categorization of cross-domain recommendation approaches

As can be seen from the previous discussion, the existing categorizations of cross-domain recommendation techniques are diverse. We aim to reconcile these categorizations in a way that captures and unifies their core ideas. For this, we focus on the exploitation of knowledge in cross-domain recommendation, which dictates the following two-level taxonomy:

- *Aggregating knowledge*. Knowledge from various source domains is aggregated to perform recommendations in a target domain (Figure 3.3a). Three use cases are considered, which will be analyzed in Section 3.4:
  - *Merging user preferences* – the aggregated knowledge consists of user preferences, e.g., ratings, tags, transaction logs, and click-through data.
  - *Mediating user modeling data* – the aggregated knowledge comes from user modeling data exploited by various recommender systems, e.g., user similarities and user neighborhoods.
  - *Combining recommendations* – the aggregated knowledge is composed of single-domain recommendations, e.g., rating estimations and rating probability distributions.
- *Linking and transferring knowledge*. Knowledge linkage or transfer between domains is established to support recommendations (Figure 3.3b). Three variants are considered, which will be analyzed in Section 3.5:
  - *Linking domains* – linking domains by a common knowledge, e.g., item attributes, association rules, semantic networks, and inter-domain correlations.
  - *Sharing latent features* – the source and target domains are related by means of implicit latent features.
  - *Transferring rating patterns* – explicit or implicit rating patterns from source domains are exploited in the target domain.

3.4 KNOWLEDGE AGGREGATION FOR CROSS-DOMAIN RECOMMENDATIONS

In this section we survey cross-domain recommendation approaches that aggregate knowledge from source domains to perform or improve recommendations in a target domain. The aggregated knowledge can be obtained at any stage of the recommendation process. In particular, it can be obtained from user preferences acquired at the user modeling stage (Section 3.4.1), from intermediate user modeling data utilized at the item relevance estimation stage (Section 3.4.2), or from item relevance estimations used at the recommendation generation stage (Section 3.4.3).

3.4.1 *Merging single-domain user preferences*

Merging user preferences from different source domains is among the most widely used strategies for cross-system personalization, and the most direct way to address the cross-domain recommendation problem (see Figure 3.4).

Research has shown that richer profiles can be generated for users when multiple sources of personal preferences are combined, revealing tastes and interests not captured in isolated domains (Abel et al., 2013; Szomszor et al., 2008b). It has been also shown that enriching sparse user preference data in a certain domain by adding user preference data from other domains, can significantly improve the generated recommendations under cold start and sparsity conditions (Sahebi and Brusilovsky, 2013; Shapira et al., 2013). These benefits, however, are accompanied by the need for having a significant amount of user preferences in multiple domains, and methods for accessing and merging the user profiles from different systems, which may have distinct types and/or representations of user preferences.

Aggregating user ratings into a multi-domain rating matrix

The most favorable scenario for aggregation-based methods implies that different systems share user preferences of the same type and representation. This scenario was addressed by Berkovsky et al. with a mediation strategy for cross-domain CF (Berkovsky et al., 2007a,b). The authors considered a domain-distributed setting where a global rating matrix $\mathbf{R}$ is split, so that single-domain recommenders store local rating matrices $\mathbf{R}_d$ having the same structure. In this setting, a target domain recommender imports rating matrices $\mathbf{R}_d$ from the source domains, integrates the local and remote rating data into the unified rating matrix R, and applies CF to R. Note that this approach can be seen as a centralized CF with ratings split across multiple domains. Nonetheless, in this approach, smaller rating matrices are more efficiently maintained by local systems, and the data is shared with the target system only when needed.

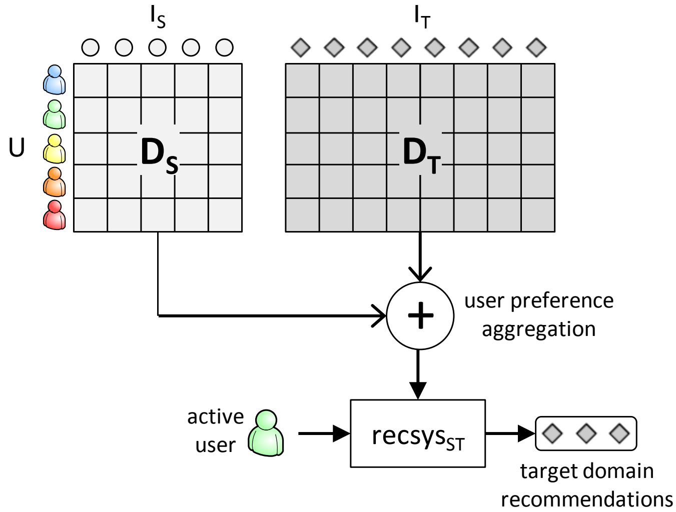

Figure 3.4: *Merging user preferences.* Data sources from different domains are merged, and a traditional single-domain recommender system is used on the merged data.

Berkovsky et al., (2007a,b) showed an improvement in the accuracy of target domain recommendations when aggregating ratings from several domains. This was also observed by Winoto and Tang, (2008). The authors collected ratings for items in several domains and conducted a study that revealed that even when there exists significant overlap and correlation between domains, recommendation accuracy in the target domain is higher if only ratings in such domain are used. Despite these findings, Winoto and Tang stated that cross-domain recommendations may have alternative benefits, in particular, serendipity and diversity.

Apart from serendipity and diversity, other benefits of cross-domain recommendations have been identified. Sahebi and Brusilovsky, (2013) examined the impact of the size of user profiles in the source and target domains on the quality of CF, and showed that aggregating ratings from several domains allows increasing the accuracy of recommendations in the target domain under cold start conditions. Similarly, Shapira et al., (2013) showed significant accuracy improvements by using aggregation-based methods when the available user preferences are sparse. In this case, the authors used a dataset composed of unary Facebook *likes* as user preferences.

Using a common representation of user preferences from multiple domains

Beyond numeric ratings and unary/binary data, other types of user preferences have also been aggregated for cross-domain recommendations. In particular, several studies have focused on aggregating user profiles composed of social tags and semantic concepts. In this context, there is no need for user or item overlap between domains, since tags and concepts are used as a common representation to merge user preferences from multiple domains.

Szomszor et al., (2008a) were among the first to correlate user profiles with *social tags* from multiple systems. They presented an architecture that transforms a set of raw tags into a set of filtered tags aligned between folksonomies in different domains. Crossing social-tag based profiles from the Delicious and Flickr folksonomies, the authors showed that filtered tags increase the overlap between domains, and allows discovering prominent user interests, locations, and events. In a follow-up work (Szomszor et al., 2008b), they extended their framework to map social tags to Wikipedia concepts, and build cross-domain user profiles composed of Wikipedia categories. An evaluation showed that new concepts of interest were learned when expanding a user tag cloud with an external repository. Related to these works, Abel et al., (2011) investigated the aggregation of a user's tag clouds from multiple systems. They evaluated a number of methods for semantic enrichment of tag overlap between domains, via tag similarities and via association rules deduced from the tagging data across systems. Aiming to analyze commonalities and differences among tag-based profiles, Abel et al., (2013) mapped tags to WordNet categories and DBpedia concepts. They used the mapped tags to build category-based user profiles, which revealed significantly more information about the users than the profiles available in specific systems. Also in the context of tag-based user profile aggregation, Fernández-Tobías et al., (2013) presented an approach that maps tags to emotional categories, under the assumption that emotions evoked by items in an entertainment domain can be represented through tags of folksonomies in which the items are annotated. Hence, emotions assigned to preferred items would be the bridge to merge user profiles across domains.

Regarding the use of *semantic concepts* as user preferences, Loizou, (2009) presented an approach that builds a graph where the nodes are associated with Wikipedia concepts describing items liked by the users, and the edges encode the semantic relationships between those concepts, obtained by integrating user ratings and Wikipedia hyperlinks. Using such a graph, a Markov chain model was used to produce recommendations by assessing the probability of traversing the graph towards a particular item, using the nodes in the user's profile as starting points. A related approach was studied by Fernández-Tobías et al., (2011) and by Kaminskas et al., (2013). The authors presented a knowledge-based framework of semantic networks that link concepts from different domains. These networks are weighted graphs, in which nodes with no incoming edges represent concepts belonging to the source domain, and nodes with no outgoing nodes represent concepts belonging to the target domain. The framework provides an algorithm that propagates the node weights, in order to identify target concepts that are most related to the source concepts. Implemented on top of DBpedia, the framework was evaluated for recommending music suited to places of interest, which were related through concepts from several domains and contextual dimensions of location and time.

Linking user preferences via a multi-domain graph

Instead of aggregating user preferences directly, several researches have focused on directed weighted graphs that link user preferences from multiple domains. Nakatsuji et al., (2010) presented an approach that builds domain-specific user graphs whose nodes are associated with users, and whose edges reflect rating-based user similarities. Domain graphs are connected via users who either rated items from several domains or shared social connections, to create a cross-domain user graph. Over this graph, a random walk algorithm retrieves items most liked by the users associated with the extracted nodes. Cremonesi et al., (2011b) built a graph whose nodes are associated with items and whose edges reflect rating-based item similarities. In this case, the inter-domain connections are the edges between pairs of items in different domains. The authors also proposed to enhance inter-domain edges by discovering new edges and strengthening existing ones, through strategies based on the transitive closure. Using the built multi-domain graph, several neighborhood- and latent factor-based CF techniques were evaluated. Tiroshi et al., (2013), collected a dataset containing user preferences in multiple domains extracted from social network profiles. The data was merged into a bipartite user-item graph, and various statistical and graph-based features of users and items were extracted from the graph. These features were exploited by a machine learning algorithm that addressed the recommendation problem as a binary classification problem.

Mapping user preferences to domain-independent features

The last type of cross-domain recommendation based on user preference aggregation is formed by the approaches that map user preferences from multiple domains to domain-independent features, and use the mapped feature-based profiles to build machine learning models that predict a user's preferences in the target domain. Although not conducting evaluations, González et al., (2005) proposed an approach for unifying single-domain user models by interoperability and coordination of several agents. In addition to user tastes and interests, the unified model is composed of the user's *socio-demographic attributes* and *emotional features*. Focusing on *personality traits*, Cantador et al., (2013) studied the relations that exist between personality types and user preferences in multiple entertainment domains, namely movies, TV, music, and books. They analyzed a large number of Facebook user profiles composed of both Big Five personality trait scores (Costa and McCrae, 1992) and explicit preferences for 16 genres in each of the above domains. As a result, the authors inferred similarities between personality-based user stereotypes in different domains. Finally, Loni et al., (2014) presented an approach that encodes rating matrices from multiple domains as real-valued feature vectors. With these vectors, an algorithm based on factorization ma-

Table 3.3: Summary of cross-domain user modeling and recommendation approaches based on merging single-domain user preferences.

Cross-domain approach	Inter-domain relationships	References
Aggregating user ratings into a single multi-domain rating matrix	Rating correlations	Berkovsky et al., 2007b UI Sahebi and Brusilovsky, 2013 U Shapira et al., 2013 U
	Rating correlations and relations between domain categories	Winoto and Tang, 2008 U
Using a common representation for user preferences from multiple domains	Social tag overlap	Szomszor et al., 2008a N Szomszor et al., 2008b N Abel et al., 2011 N Abel et al., 2013 N Fernández-Tobías et al., 2013 N
	Semantic relationships between domain concepts	Loizou, 2009 N Fernández-Tobías et al., 2011 N Kaminskas et al., 2013 N
Linking user preferences via a multi-domain graph	Rating-based user/item similarities	Nakatsuji et al., 2010 U Cremonesi et al., 2011b U
	Patterns of user-item graph-based features	Tiroshi et al., 2013 U
Mapping user preferences to domain-independent features	Socio-demographic and emotional features	González et al., 2005 N
	Personality features	Cantador et al., 2013 N
	User-item interaction features	Loni et al., 2014 U

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap

chines (Rendle, 2012) finds patterns between features from the source and target domains, and outputs preference estimations associated with the input vectors.

We summarize the discussed aggregation-based methods in Table 3.3. Aggregating ratings from several CF systems is the simplest method, but requires access to user profiles, and a significant rating overlap between domains, which may not be achievable in real situations. Thus, most aggregation-based methods transform user preferences from multiple domains into a common representation, independent of the domains of interest, and usable for establishing inter-domain data relations and overlaps. For this purpose, social tags and semantic concepts serve as the main types of user preferences. More recent methods focus on aggregating several sources of user preferences from multiple domains into a single graph. Due to the increasing use of social media, we envision that novel cross-domain recommendation approaches that both unify user preferences and aggregate them into multi-domain graphs will be developed.

3.4.2 *Mediating single-domain user modeling data*

Not only immediate user preferences, but also other recommendation-related information about users, items, and domains may be aggregated or mediated (see Figure 3.5). An early approach for cross-domain recommendation through mediation was proposed by Berkovsky et al., (2008). The central idea behind user model mediation is that importing any user modeling data from source recommenders may benefit a target recommender (Berkovsky et al., 2005) – the mediation can enrich the user models of the target recommender, and yield more accurate recommendations. What data can be mediated between the source and the target recommenders? The most simple scenario covered in Section 3.4.1 includes importing the user models, whereas more complex scenarios include mediating specific recommendation data.

Aggregating neighborhoods to generate recommendations

For example, in a CF system, cross-domain mediation may import the list of nearest neighbors. This is underpinned by two assumptions: (i) there is overlap of users between domains, and (ii) user similarity spans across domains, i.e., if two users are similar in a source domain, they are similar also in the target domain. This idea was leveraged in the heuristic variant of cross-domain mediation developed by Berkovsky et al., (2007b). There, it was shown that importing nearest neighbors, and computing their similarity with the target domain data only, can produce more accurate recommendations than single-domain recommendations. A similar idea was formulated by Shapira et al., (2013) as the k nearest neighbors source aggregation. They used multi-domain Facebook data to produce the set of candidate nearest neighbors, and compute their local similarity degree in the source domain. This allowed overcoming the new user problem and the lack of ratings in the target domain. Another attempt to use multi-domain Facebook data was done by Tiroshi and Kuflik, (2012). They applied random walks to identify source domain-specific neighbor sets, which were used to generate recommendations in the target domain.

Aggregating user-to-user similarities to generate recommendation

Aggregating the lists of nearest neighbors relies on their data in the target domain only, which may be too sparse and result in noisy recommendations. Thus, one could consider importing and aggregating also the degree of their similarity in the source domain. This approach was referred to in (Berkovsky et al., 2007b) as cross-domain mediation. A content-based and a statistical variant of domain distance metrics were evaluated in (Berkovsky et al., 2006), producing comparable results and outperforming single-domain recommendations. The weighted k-NN aggregation was further enhanced by Shapira et al., (2013). The authors compared several weighting schemes, the performance of which was consistent across several metrics and recommendation tasks.

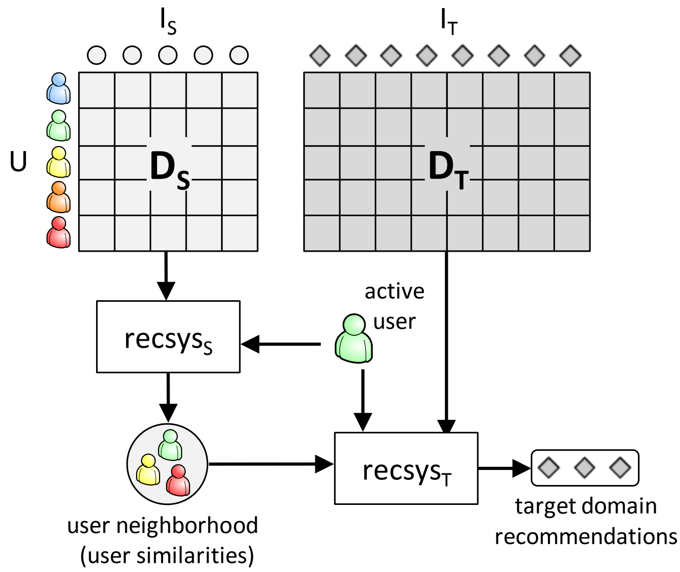

Figure 3.5: *Mediating user modeling data*. A model is learned in the source domain (e.g., the neighborhood of a user) and used in the target domain.

Exploiting user neighborhoods to enhance target domain user models

The above scenarios of cross-domain mediation assume an overlap in the sets of users. An analogous scenario refers to a setting where items overlap between the source and target domains, which opens the opportunity for further mediation. One of them, involving only the music domain, but two systems (for tagging and for blogging) was studied by Stewart et al., (2009). The authors leveraged the tags assigned by similar users on Last.fm in order to recommend tags on Blogger.

Combining probabilistic user models

Moving from CF to latent factor-based methods, we highlight two works compatible with the user modeling data mediation pattern. Low et al., (2011) developed a hierarchical probabilistic model that combines user information across multiple domains, and facilitates personalization in domains with no prior user interactions. The model is underpinned by a global user profile based on a latent vector, and a set of domain-specific latent factors that eliminate the need for common items or features.

Combining heterogeneous user preferences

Pan et al., (2012) dealt with transferring uncertain ratings, i.e., expected rating range or distribution derived from behavioral logs, using latent features of both users and items. The uncertain ratings were transferred from the source to the target domain, and leveraged there as constraints for the matrix factorization model.

We summarize the mediation-based approaches in Table 3.4. As can be seen, they all imply either user- or item-overlap between the source and target domains. These are necessary for identifying high-level user preferences spanning across domains. This often requires sharing of user data between several systems, which is avoided due to commercial competition and conflicts with privacy regulations. How-

Table 3.4: Summary of cross-domain recommendation approaches based on mediating single-domain user modeling data.

Cross-domain approach	**Inter-domain relationships**	**References**
Aggregating neighborhoods to generate recommendations	Rating-based user similarities	Berkovsky et al., 2007b U Tiroshi and Kuflik, 2012 UI Shapira et al., 2013 U
Aggregating user-to-user similarities to generate recommendations	Content- and rating-based user similarities	Berkovsky et al., 2007b U Shapira et al., 2013 U
Exploiting user neighborhoods to enhance target domain user models	User overlap	Stewart et al., 2009 I
Combining probabilistic user models	Latent features of domains and global user preferences	Low et al., 2011 U
Combining heterogeneous user preferences	Domain-dependent constraints on matrix factorization	Pan et al., 2012 UI

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap.

ever, it is usual for a user to utilize multiple systems (or, in a more common use-case, to have accounts on multiple social networks), and thus cross-domain recommendations through mediation is a feasible scenario. Most of the surveyed approaches apply simple mediation methods, whereas the last two are based on latent representations, and apply probabilistic or transfer learning models. None of these works counts on explicit domain distance or similarity, which will be elaborated in Section 3.5.1). Hence, we conjecture that more future work will address cross-domain recommendation by mediating richer user modeling data.

3.4.3 *Combining single-domain recommendations*

Overlap of both users and items allows aggregating ready-made single-domain recommendations (see Figure 3.6). Contrarily to the mediation-based cross-domain recommendation scenarios, the predicted recommendations from the source domain may inform on their own to the target domain recommender. Hence, the central question in combining single-domain recommendation refers to the weights assigned to recommendations coming from the source domains, which reflect their importance for the target domain. These weights may

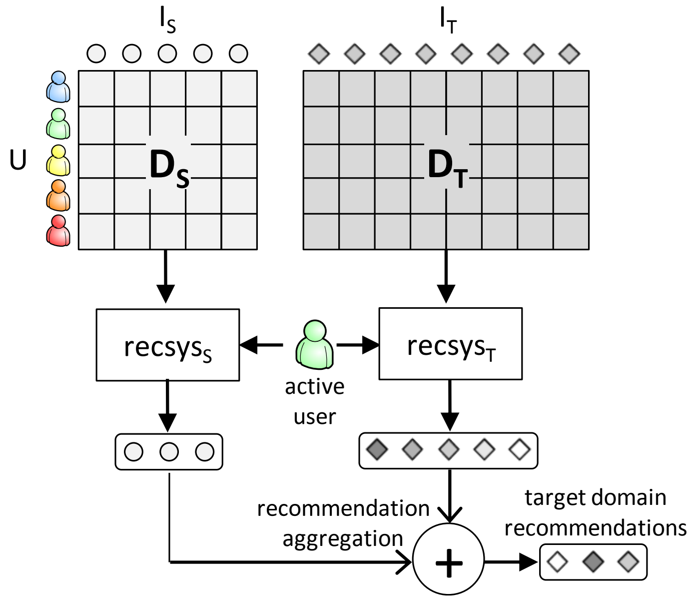

Figure 3.6: *Combining single-domain recommendations*. Recommendations are generated independently for each domain and later merged for the final recommendation.

be computed through various factors, such as the reliability of each recommender, distance between the domains, and so forth.

Aggregating user rating predictions

The idea of combining single-domain recommendations was referred to in (Berkovsky et al., 2007a,b) as remote-average mediation. There, movie ratings were partitioned into domains according to the genres of the movies. Since movies combine elements from multiple genres, and users watch movies from various genres, the user- and item-overlap are both present. This allows computing stand-alone recommendations in the source domains, and aggregating them for the target domain. Weighted aggregation of single-domain recommendations also was studied by Givon and Lavrenko, (2009). The authors focused on the book recommendation task, accomplished using two different methods. Standard CF recommendations were complemented by relevance model-based recommendations, relying on the similarity of a book and the user's model, both consisting of book contents and tags assigned to the book. The two were combined in a weighted manner, such that the relative importance of the CF recommendations increased with the number of ratings available.

Combining estimations of rating distributions

A relevant approach for cross-domain consensus regularization, although applied to classification problems and not to recommender systems, was proposed by Zhuang et al., (2010). The central contribution of that work is a framework for learning from multiple source domains, and reconciling discrepancies between the classifiers using the local data of the target domain. One of the advantages of the framework is that it does not require overlaps in either the user or item sets.

Table 3.5: Summary of cross-domain recommendation approaches based on combining recommendations from single-domain user preferences.

Cross-domain approach	Inter-domain relationships	References
Aggregating user rating predictions	Rating-based user similarities	Berkovsky et al., 2007b UI Givon and Lavrenko, 2009 UI
Combining estimations of rating distribution	Rating distribution similarities	Zhuang et al., 2010 N

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap.

The revised approaches that combine single-domain recommendations are summarized in Table 3.5. Clearly, the key point for this group of cross-domain recommenders refers to the way the standalone source domain recommendations are combined. This is touched upon in (Zhuang et al., 2010), but also addressed in numerous researches outside the recommender systems space. It should be highlighted that the single-domain recommenders can use various techniques, and the combination of their outputs is independent of other components, e.g., user modeling, contextualization, and presentation, which makes this cross-domain aggregation variant attractive for practical recommenders.

3.5 KNOWLEDGE LINKAGE AND TRANSFER FOR CROSS-DOMAIN RECOMMENDATION

In this section we survey cross-domain recommendation approaches that link or transfer knowledge between domains, enhancing the information available in the target domain for the generation of recommendations. The knowledge linkage and transfer can be done explicitly —e.g., via common item attributes, semantic networks, association rules, and inter-domain user preference similarities (Section 3.5.1) –, implicitly by means of latent features shared by domains (Section 3.5.2), or by means of rating patterns transferred between domains (Section 3.5.3).

3.5.1 *Linking domains*

A natural approach to address the heterogeneity of several domains is to identify correspondences between their characteristics. For instance, we may link a particular movie and a book because both belong to genres that can be semantically mapped, e.g., comedy movies and humorous books. In general, such inter-domain correspondences may be established directly using some kind of common knowledge

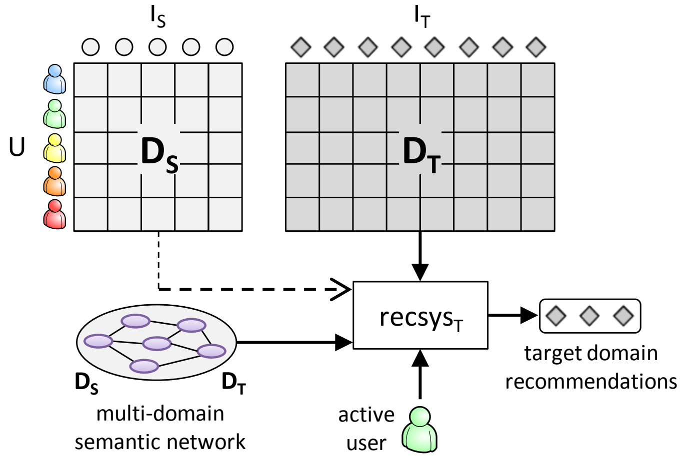

Figure 3.7: *Linking domains.* An external knowledge source is used to link items from different domains. User preferences in the source domain may be used to adapt the item linkage.

between domains, e.g., item attributes, semantic networks, association rules, and inter-domain preference-based similarities or correlations (see Figure 3.7).

Relating and filtering items via common attributes

These links are valuable sources of information for reasoning across domains. A recommender system could identify potentially relevant items in the target domain by selecting those that are related to others in the source domains, and for which the user has expressed a preference in the past. Besides, inter-domain similarities and correlations can be exploited to adapt or combine knowledge transferred from different domains. One of the earliest approaches for linking domains was explored by Chung et al., (2007). Aiming to support the decision making process in recommendation, they proposed a framework for designing personalized filtering strategies. In the framework, relevant items in the target domain are selected according to the attributes they have in common with items in the source domain the user is interested in. That is, the inter-domain links are established through the overlap of item attributes, and no user or item overlap between the domains is required.

Building semantic network linking domain concepts

Conversely to the use case of (Chung et al., 2007), in a realistic setting, items are highly heterogeneous, and often no common attributes between domains can be found. To address this situation, we may establish more complex, likely indirect relations between items in different domains. Hence, when suitable knowledge repositories are available, concepts from several domains can be connected by the means of semantic properties, forming semantic networks that explicitly link the domains of interest. Along these lines, Loizou, (2009) proposed to use Wikipedia as a universal vocabulary to express and relate user preferences across multiple domains. The author presented an approach that builds a graph, the nodes of which represent concepts (Wikipedia pages) describing items liked by the users, and edges encode the semantic relationships between those concepts, obtained by integrating user ratings and Wikipedia hyperlinks. Using such a graph, a Markov chain model produces recommendations by assessing the probability of traversing the graph from the nodes in the user's profile as a starting point toward the recommendable items.

A major difficulty of the above approaches is the well known *knowledge acquisition* problem, that is, building the above mentioned knowledge repositories. To address this problem, information has to be extracted and stored in a formal and structured representation that can be exploited by a recommender. Fernández-Tobías et al., (2011) and Kaminskas et al., (2013) envisioned Linked Data as a solution to the problem. Specifically, they proposed a framework for extracting a multi-domain semantic network from the DBpedia ontology, which links items and concepts in the source and target domains. Over the extracted network, a constrained spreading algorithm computes semantic similarities to rank and filter items in the target domain.

Relating item types via knowledge-based rules

Inter-domain association rules have also been explored as an alternative to relate various types of items. In this direction, Azak, (2010) presented a framework for cross-domain recommendation in which knowledge-based rules defined by domain experts facilitate mapping between attributes in distinct domains, e.g., "people who like romance drama movies also like dramatic poetry books." These rules are then used to enhance CB and CF recommendations, adjusting the predicted ratings whenever rule conditions hold. Cantador et al., (2013) related user personality types with domain-dependent preferences by means of automatically generated association rules. The authors also extracted personality stereotypes for sets of domain genres. Based on these stereotypes, inter-domain similarities were computed between genres, which may be used to support knowledge transfer between domains.

Computing inter-domain similarities

Instead of linking domains by mapping attributes, an alternative way to transfer knowledge is to compute similarities or correlations between domains based on user preference or item content analysis. In an early work, Berkovsky et al., (2006) explored this idea aiming to identify related domains, from which user data would be imported and utilized to enrich the user model in the target domain. The proposed approach makes use of web directories to identify websites that characterize the domains of interest. Then, the approach establishes domain similarities by computing the cosine similarity between the TF-IDF term vectors of the domains' websites. We note that this method requires no overlap of users or items, but rather an external source of representative documents classified to several domains.

Constraining matrix factorization with inter-domain similarities

Another way of exploiting inter-domain similarities for cross-domain recommendation consists of integrating them into the probabilistic matrix factorization method (Salakhutdinov and Mnih, 2007). Specifically, such similarities are imposed as constraints over user or

Table 3.6: Summary of cross-domain user modeling and recommendation approaches based on linking domains.

Cross-domain approach	**Inter-domain relationships**	**References**
Relating and filtering items via common attributes	Item attribute overlap	Chung et al., 2007 N
Building semantic network linking domain concepts	Semantic relationships between domain concepts	Loizou, 2009 N Fernández-Tobías et al., 2011 N Kaminskas et al., 2013 N
Relating item types via knowledge-based rules	Inter-domain knowledge-based rules	Azak, 2010 N Cantador et al., 2013 N
Computing inter-domain similarities	Text overlap	Berkovsky et al., 2006 N
Constraining matrix factorization with inter-domain similarities	Rating overlap	Cao et al., 2010 U Zhang et al., 2010 N
	Social tag overlap	Shi et al., 2011 N

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap.

item latent factors when jointly factorizing rating matrices. For instance, Cao et al., (2010) proposed an approach in which inter-domain similarities are implicitly learned from data, as model parameters in a non-parametric Bayesian framework. Since user feedback is used to estimate the similarities, user overlap between the domains is required. Addressing the sparsity problem, Zhang et al., (2010) adapted the probabilistic matrix factorization method to include a probability distribution of user latent factors that encodes inter-domain correlations. One strength of this approach is that user latent factors shared across domains are not needed, allowing more flexibility in capturing the heterogeneity of domains. Instead of automatically learning implicit correlations in the data, Shi et al., (2011) argued that explicit common information is more effective, and relied on shared social tags to compute cross-domain user-to-user and item-to-item similarities. Similarly to previous approaches, rating matrices from the source and target domains are jointly factorized; but in this case user and item latent factors from each domain are restricted, so that their product is consistent with the tag-based similarities.

We have reviewed approaches that establish links and compute similarities between domains, which are summarized in Table 3.6. We observe that the majority of the proposed methods do not require inter-domain user or item overlap. Instead, linking approaches exploit content information to establish the inter-domain relationships. Likewise, in (Berkovsky et al., 2006; Shi et al., 2011), similarities are computed

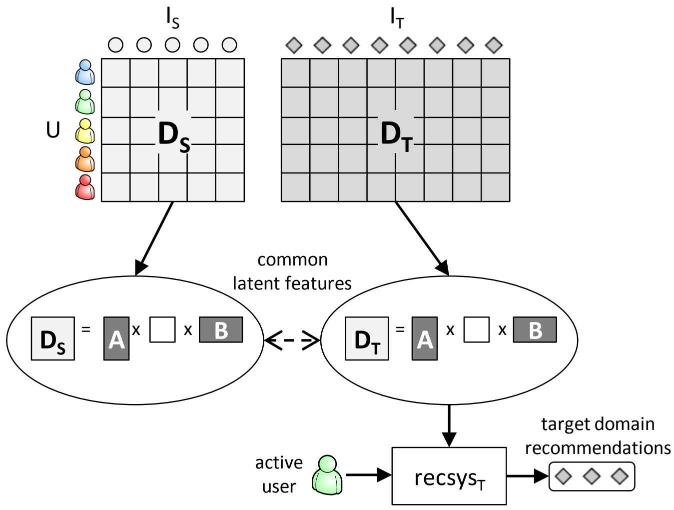

Figure 3.8: *Sharing latent features*. Latent features models are learned simultaneously on both the source and target domains, constraining user and/or item features to be the same across the domains.

based on common text and social tags. For these approaches, it is also worth noticing that no one clearly outperforms the others, since most of them are designed for particular cross-domain scenarios and, to the best of our knowledge, have not been compared empirically.

3.5.2 *Sharing latent features by domains*

Latent factor models are among the most popular CF techniques (Koren, 2008). In these models, user preferences and item attributes, which are typically very sparse, are characterized through a reduced set of latent factors discovered from data, to obtain a denser representation. The assumption is that using the new representation, latent user preferences and item attributes are better captured and matched.

Related to the *what to transfer* aspect of transfer learning (Pan and Yang, 2010), latent factors shared between domains can be exploited to support cross-domain recommendations (see Figure 3.8). Also, as pointed in Section 3.3, two types of approaches have been studied to address the *how to transfer* aspect; namely, *adaptive* and *collective* models. In the former, latent factors are learned in the source domain, and are integrated into a recommendation model in the target domain, while in the latter, latent factors are learned simultaneously optimizing an objective function that involves both domains.

Using user and item latent features of source domains to regularize latent features in a target domain

Pan et al., (2010) addressed the sparsity problem in the target domain following the adaptive approach, proposing to exploit user and item information from auxiliary domains where user feedback may be represented differently. In particular, they studied the case in which users express binary like/dislike preferences in the source domain, and utilize 1-5 ratings in the target domain. Their approach performs singular value decomposition (SVD) in each auxiliary domain, in order to separately compute user and item latent factors, which are then shared with the target domain. Specifically, transferred factors are integrated into the factorization of the rating matrix in the target domain and added as regularization terms so that specific characteristics of the target domain can be captured.

Using the same latent factors to jointly factorize the rating matrices in the source and target domains

Latent factors can also be shared in a collective way, as studied by Pan et al., (2011). In this case, instead of learning latent features from the source domains and transferring them to the target domain, the authors proposed to learn the latent features simultaneously in all the domains. Both user and item factors are assumed to generate the observed ratings in every domain, and, thus, their corresponding random variables are shared between the probabilistic factorization models of each rating matrix. Moreover, the factorization method is further extended by incorporating another set of factors that capture domain-dependent information, resulting in a tri-factorization scheme. A limitation of the proposed approach is that the users and items from the source and target domains have to be identical.

Extending matrix factorization with a vector of latent factors associated to social tags

Instead of focusing on sharing latent factors, Enrich et al., (2013), and Fernández-Tobías and Cantador, (2014) studied the influence of social tags on rating prediction, as a knowledge transfer approach for cross-domain recommendations. The authors presented a number of models based on the SVD++ algorithm (Koren, 2008) to incorporate the effect of tag assignments into rating estimation. The underlying hypothesis is that information about item annotation in a source domain can be exploited to improve rating prediction in a target domain, as long as a set of common tags between the domains exists. In the proposed models, tag factors are added to the latent item vectors, and are combined with user latent features to compute rating estimations. The difference between these models is in the set of tags considered for rating prediction. In all the models knowledge transfer is performed through the shared tag factors in a collective way, since these are computed jointly for the source and the target domains.

Sharing latent features via a user-item-domain tensor factorization

Hu et al., (2013) presented a more complex approach that takes domain factors into account. There, the authors argue that user-item dyadic data cannot fully capture the heterogeneity of items, and that modeling domain-specific information is essential to make accurate predictions in a setting where users typically express their preferences in a single domain. They referred to this problem as the *unacquainted world*, and proposed a tensor factorization algorithm to exploit the triadic user-item-domain data. In that method, rating matrices from several domains are simultaneously decomposed into shared user, item, and domain latent factors, and a genetic algorithm automatically estimates optimal weights of the domains.

Table 3.7 summarizes the described approaches sharing latent factors across domains. In contrast to the methods presented in Sec-

Table 3.7: Summary of cross-domain recommendation approaches based on latent features shared by domains.

Cross-domain approach	Inter-domain relationships	References
Using user and item latent features of source domains to regularize latent features in a target domain	Shared latent user preferences and latent item attributes	Pan et al., 2010 [UI]
Using the same latent factors to jointly factorize the rating matrices in the source and target domains	User and item overlap	Pan et al., 2011 [UI]
Extending matrix factorization with a vector of latent factors associated to social tags	Social tag overlap	Enrich et al., 2013 [N] Fernández-Tobías and Cantador, 2014 [N]
Sharing latent features via a user-item-domain tensor factorization	Rating overlap	Hu et al., 2013 [U]

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap.

tion 3.5.1, these approaches require inter-domain user or item overlap to extract shared latent factors, unless shared content information is available (Enrich et al., 2013; Fernández-Tobías and Cantador, 2014). As in the previous section, it is worth noticing the lack of a comparative study of the approaches. Again, the reason for this may be that the considered cross-domain task and data overlap scenarios vary among works.

3.5.3 *Transferring rating patterns between domains*

Rather than sharing user or item latent factors for knowledge transfer, a different set of approaches analyzes the structure of rating data at the community level. These methods are based on the hypothesis that even when their users and items are different, close domains are likely to have user preferences sampled with the same population. Therefore, latent correlations may exist between preferences of groups of users for groups of items, which are referred to as *rating patterns*. In this context, rating patterns can act as a bridge that relates the domains (see Figure 3.9), such that knowledge transfer can be performed in either adaptive or collective manners. In the adaptive setting, rating patterns are extracted from a dense source domain. In the collective setting, data from all the domains are pulled together

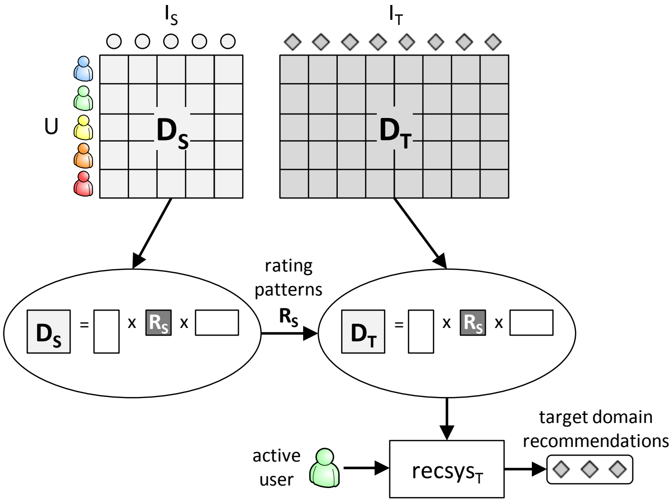

Figure 3.9: *Transferring rating patterns*. A co-clustering model is learned on the source domain to obtain rating patterns, which are used to cluster users and items in the target domain.

and jointly exploited, even though users and items do not overlap across domains.

Extracting association rules from user rating behaviour

Lee and Seung, (2001) proposed one of the first approaches to exploit rating patterns for cross-domain recommendation. Similarly to the cross-domain mediation proposed by Berkovsky et al., (2007a), global nearest neighbors are identified by adding the similarity scores from each domain. Then, patterns of items commonly rated together by a set of neighbors are discovered using association rules. Finally, in the recommendation stage, rating predictions are computed with the standard user-based CF algorithm, but enhanced with the user's rules that contain the target items.

Transferring implicit cluster-level rating patterns between domains

Li et al., (2009a) proposed an adaptive method based on simultaneously co-clustering users and items in the source domain to extract rating patterns. Clustering is performed using a tri-factorization of the source rating matrix Ding et al., 2006. Then, knowledge is transferred through a *codebook*, a compact cluster-level matrix computed in the source domain taking the average rating of each user-item cluster. In the target domain, missing ratings are predicted using the codebook. Moreno et al., (2012) extended the codebook idea to a scenario in which various source domains contribute to the target domain. The approach is based on a linear combination of codebooks, where the weights are learned by minimizing the prediction error in the target domain.

In a related work, Li et al., (2009b) extended the same idea to a collective approach using a probabilistic framework. Instead of relying on an dense source domain data to build the codebook, all rating matrices are pulled together to extract the shared patterns. Furthermore, rather than having each user/item belonging to a single cluster, a probability distribution is introduced to allow users and items be-

Table 3.8: Summary of cross-domain recommendation approaches based on transferring rating patterns between domains.

Cross-domain approach	**Inter-domain relationships**	**References**
Extracting association rules from user rating behavior	Rating overlap	Lee and Seung, 2001 [U]
Transferring implicit cluster-level rating patterns between domains	Rating patterns	Li et al., 2009a,b [N] Moreno et al., 2012 [N] Cremonesi and Quadrana, 2014 [N]
	Domain-independent parts of rating patterns	Gao et al., 2013 [N]

(N) no overlap, (U) user overlap, (I) item overlap, (UI) user and item overlap.

long to multiple clusters, with distinct membership degrees. In the same fashion, the ratings associated with each user-item cluster are also given by a conditional probability distribution. In this way, a generative rating model is obtained, since the ratings of each domain can be recovered by drawing users and items from the shared cluster-level model, and then drawing the expected rating conditioned to the user-item cluster.

A strength of both approaches is that neither overlap of users nor of items is required. However, Cremonesi and Quadrana, (2014) empirically refuted the *codebook* method, showing that it does not transfer any knowledge when source and target domains do not overlap. According to their experiments, a codebook filled with random values achieved similar accuracy to the original approach, concluding that the supposed improvement in performance was due to an artifact in the evaluation methodology.

Finally, Gao et al., (2013) followed the idea of extracting rating patterns by co-clustering rating matrices, and addressed two limitations of previous methods. First, they argued that some domains are more related to the target domain than others, and this cannot be captured using identical rating patterns. Second, they hypothesized that performance may suffer when the domains are diverse, and do not share common rating patterns. To overcome these limitations, the authors proposed a model capable of controlling the amount of knowledge transferred from each domain. Specifically, they used a co-clustering algorithm of (Li et al., 2009a), but split the extracted rating patterns into a shared part and a domain-specific part. In contrast to (Li et al., 2009a), optimization is performed in a collective way, since the shared part of the rating patterns is learned simultaneously from all the domains.

Table 3.8 summarizes the described cross-domain approaches based on transferring rating patterns between domains. We observe that

Table 3.9: Summary of cross-domain recommendation approaches based on the technique used to partition the data into training and test sets.

Data partitioning	**References**
Online studies	Braunhofer et al., 2013; Fernández-Tobías et al., 2013; Shapira et al., 2013; Szomszor et al., 2008b; Winoto and Tang, 2008
Leave-all-users-out	Cremonesi et al., 2011b; Goga et al., 2013; Hu et al., 2013; Jain et al., 2013; Kaminskas et al., 2013; Loni et al., 2014; Shapira et al., 2013; Tiroshi and Kuflik, 2012
Leave-some-users-out	Abel et al., 2011, 2013; Li et al., 2009a,b; Stewart et al., 2009
Hold-out	Li et al., 2011; Nakatsuji et al., 2010; Pan et al., 2008; Pan and Yang, 2013; Pan et al., 2010, 2012; Sahebi and Brusilovsky, 2013; Shi et al., 2011; Tang et al., 2011; Zhang et al., 2013; Zhang et al., 2010; Zhao et al., 2013

more recent methods based on clustering do not rely on any overlap between domains. However, as discussed in (Gao et al., 2013), care must be taken in order not to degrade performance by transferring noisy patterns from unrelated domains. We therefore conjecture that further research on the *when to transfer* aspect (Pan and Yang, 2010) will be conducted, to identify valuable information from source domains.

3.6 EVALUATION OF CROSS-DOMAIN RECOMMENDER SYSTEMS

In this section, we discuss the methodologies used to evaluate cross-domain recommender systems. The focal point is that such systems cannot be evaluated in a problem-independent way; whether a cross-domain recommender system is an appropriate solution cannot be evaluated without taking into account for what it is intended. The nature of the evaluation must be connected to the purpose for which the recommendations are required. Thus, we compare the corresponding evaluation methods based on the cross-domain recommendation goals addressed in the literature (see Section 3.2.2).

Three types of evaluations can be used to compare (cross-domain) recommender systems (see Section 2.4.1). As most works use offline experiments (with a few performing user studies, and no online trials, see Table 3.9), we focus on offline experiments. We refer the reader to Section 2.4 for an extensive discussion on methodologies and metrics used to evaluate recommender systems.

The decision regarding the evaluation method is often critical, as each one reflects a specific task or goal. Many schemes for offline evaluation exist, which differ in a number of aspects: *data partitioning*, *metrics*, and *sensitivity analysis* (e.g., relative density of domain datasets,

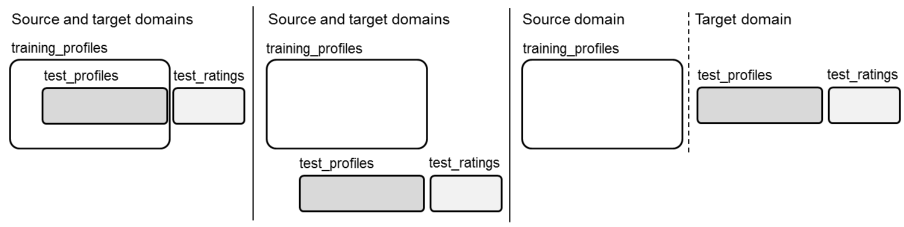

Figure 3.10: Partitioning of $\mathcal{D}$: (left) *hold-out*, test ratings are sampled and hidden without partitioning the users; (middle) *leave-some-users-out*, users are split into disjoint training/test sets; (right) *leave-all-users-out*, ratings in the target dataset used as test profiles and ratings.

and degree of overlap between domains), as discussed respectively in the next sections.

3.6.1 *Data partitioning*

In order to evaluate algorithms offline, it is necessary to simulate the process where the system makes recommendations, and users evaluate them. This requires pre-recorded datasets of interactions between users and items. In cross-domain applications, there are (at least) two potentially overlapping datasets: the source dataset $\mathcal{D}_S$ and the target dataset $\mathcal{D}_T$.

We assume $\mathcal{D}_S$ and $\mathcal{D}_T$ are chosen according to the recommendation task and goal in hand. For instance, if we are evaluating a cross-selling recommender, $\mathcal{D}_S$ and $\mathcal{D}_T$ are set at the *item level* as described in Section 3.2.1, contain items of different nature, like movies and books, and have overlapping users. On the contrary, if we are evaluating a cross-domain recommender as a tool to increase recommendation diversity, $\mathcal{D}_S$ and $\mathcal{D}_T$ are set at the *item attribute level*, with items of the same type, but differ in the value of certain attribute, as comedy and drama movies.

Training and test users and ratings

In offline evaluations, a portion of $\mathcal{D}_T$ is hidden to facilitate prediction of the available knowledge, and gauge the quality of the recommendations. There is a number of ways to choose the ratings to be hidden. The most general approach creates three subsets of ratings from the original datasets: (i) $\mathcal{D}_{\text{train}}$, which contains the set of ratings from users $\mathcal{U}_{\text{train}}$ for items $\mathcal{I}_{\text{train}}$ that are used to train the algorithms under evaluation; (ii) $\mathcal{D}_{\text{test}}$, which contains the set of users $\mathcal{U}_{\text{test}}$ and their known ratings for items $\mathcal{I}_{\text{test}}$ that are used as input profiles for the trained recommender; and (iii) $\mathcal{D}_{\text{test ratings}}$, which contains the set of users $\mathcal{U}_{\text{test}}$ and their hidden ratings for items $\mathcal{I}_{\text{test ratings}}$ that are used as the ground truth to evaluate the recommendations.

Depending on the choice of the $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{test}}$, and $\mathcal{D}_{\text{test ratings}}$ subsets, different evaluation data partitions can be designed.

- *Hold-out* (Figure 3.10, left) is implemented when $\mathcal{D}_{\text{test}} \subseteq \mathcal{D}_{\text{train}}$, i.e., test ratings are sampled and hidden from the original dataset without partitioning the users. This partition is suitable to evaluate linked- and multi-domain recommenders with the accuracy goal, and is applicable to memory-based recommenders, which are unable to provide recommendations to new users.

- *Leave-some-users-out* (Figure 3.10, middle) is implemented when $\mathcal{U}_{\text{train}} \cap \mathcal{U}_{\text{test}} = \varnothing$, i.e., the users are split into two disjoint subsets: one for training and one for testing. This partition is suitable to evaluate a cross-domain recommender with the new user goal.

- *Leave-all-users-out* (Figure 3.10, right) is implemented when $\mathcal{D}_{\text{train}} \cap \mathcal{D}_{T} = \varnothing$, i.e., the ratings in the target dataset are used only as profile and test ratings. This partition is suitable to evaluate a cross-domain recommender with the cold start and new item goals.

3.6.2 *Sensitivity analysis*

The performance of a cross-domain recommender system is mainly affected by three parameters: the overlap between the source and target domains, the size of the target user's profile, and the density of the target domain data. Thus, the evaluation of a cross-domain recommendation approach mostly considered the sensitivity of the underlying recommendation algorithm with respect to these three parameters.

Data overlap between domains

Most works have assumed an overlap of users between the source and target domains. They all conducted evaluations with 100% of overlap, except for two works. Cremonesi et al., (2011b) analyzed the behavior of various cross-domain recommenders by varying the percentage of user-overlap in the range 0%-50%, and Zhao et al., (2013) adopted a similar evaluation by varying the percentage of user overlap in the range 0%-100%. Fewer works (Berkovsky et al., 2008; Cremonesi et al., 2011b; Pan and Yang, 2013; Zhao et al., 2013) studied the case of item overlap, and they all assume to have the same catalog of items across domains. Some works (Abel et al., 2013; Braunhofer et al., 2013; Fernández-Tobías et al., 2013; Kaminskas et al., 2013; Stewart et al., 2009; Szomszor et al., 2008b) studied the case of overlapping features, especially social tags. Shi et al., (2011) studied the sensitivity of the cross-domain recommender by varying the number of overlapping tags between 5 and 50.

User profile size

Some works (Berkovsky et al., 2008; Li et al., 2009a,b; Sahebi and Brusilovsky, 2013; Shi et al., 2011) have studied the sensitivity of recommendations as a function of the user profile size, i.e., the number of ratings provided by the user receiving the recommendations. This is particularly important for the cold start and new user goals. Both

Table 3.10: Summary of variables for sensitivity analysis of cross-domain recommender systems.

Parameter	**References**
Data overlap between domains	Abel et al., 2013; Cremonesi et al., 2011b; Shi et al., 2011; Zhao et al., 2013
User profile size	Berkovsky et al., 2007a,b, 2008; Li et al., 2009a,b; Sahebi and Brusilovsky, 2013; Shi et al., 2011
Target domain data density	Cao et al., 2010; Cremonesi et al., 2011b; Pan et al., 2008; Pan et al., 2010; Shapira et al., 2013

Pan et al., (2010) and Abel et al., (2013) developed tag-based recommenders, and performed their analysis by varying the number of tags in the user profile in the 10–40 and 0–150 ranges, respectively. Others conducted a similar analysis on rating-based recommenders: Shi et al., (2011) varied the profile size from 20 to 100 ratings, Berkovsky et al., (2008) varied the profile size from 3% to 33% of ratings, and Li et al., (2009a,b) and Sahebi and Brusilovsky, (2013) varied the profile size in the range of 5–15 and 1–20 ratings, respectively.

Target domain data density

Finally, some works (Cao et al., 2010; Cremonesi et al., 2011b; Pan et al., 2010; Shapira et al., 2013) have studied the quality of recommendations as a function of the dataset density. This is important for the cold start and accuracy goals. Cao et al., (2010) varied the density of the multi-domain dataset, i.e., the union of source and target datasets, between 0.2% and 1%. Shapira et al., (2013) varied the density of the dataset between 1% and 40%, but only for the baseline single-domain algorithms, while evaluating cross-domain algorithms at the 1% density. Cremonesi et al., (2011b) varied the density of the target domain between 0.1% and 0.9%. The sensitivity analyses performed in the above works are summarized in Table 3.10.

3.7 SUMMARY

The proliferation of e-commerce sites and online social media has allowed users to provide preference feedback and maintain profiles in multiple systems, reflecting a variety of their tastes, interests and needs. Leveraging all the user preferences available in several systems or domains may be beneficial for generating more encompassing user models and better recommendations, e.g. through mitigating the cold start and sparsity problems in a target domain, or enabling personalized cross-selling recommendations for items from multiple domains.

In this context, cross-domain recommendation is an emerging topic with plenty of research opportunities. Numerous approaches have been proposed from multiple perspectives in several areas. In this chapter we have provided a comprehensive formalization of the problem and of the definition of domain, based on the level of granularity considered to distinguish between them. We have also identified the main tasks and goals addressed in previous works, and have presented an exhaustive categorization of the used recommendation techniques, mainly into knowledge aggregation approaches and knowledge linkage/transfer approaches. Finally, we have reviewed the different strategies used to evaluate cross-domain recommender systems in the state of the art, identifying some works that analyze the performance in terms of the number of available target domain preferences, which, as we shall see in the following chapters, is the basis for the cold start evaluation of the cross-domain recommendation models presented in this thesis.

Part II

PROPOSED SOLUTIONS

4 EXPLOITING SOCIAL TAGS IN MATRIX FACTORIZATION FOR CROSS-DOMAIN COLLABORATIVE FILTERING

During the last years, there have been increasing success and popularization of social tagging services. In these services, users create or upload contents, annotate the contents with freely chosen words –referred as *tags*–, and share both contents and tags with others. The nature of tagged contents is manifold, e.g., photos in Flickr, music tracks in Last.fm, videos in YouTube, and websites in Delicious, to name a few. The whole set of tags in each system constitutes a collaborative, unstructured knowledge classification scheme, commonly known as folksonomy. This implicit classification can be considered as a source of user preferences, since users assign tags to own contents and contents they like from others, and thus can be used for recommendation purposes.

In this chapter we present our first matrix factorization model for cross-domain collaborative filtering, which exploits social tags as user preferences shared between different domains. In Section 4.1 we provide motivations to exploit social tags for cross-domain recommendation, and introduce the bases of our model. In Section 4.2 we review state of the art approaches that utilize tags for recommendation, focusing on those based on matrix factorization to support cross-domain recommendation, and in Section 4.3 we present our recommendation model. Next, in Section 4.4 we describe the experiments conducted to evaluate the model, and in Section 4.5 we discuss the results achieved. Finally, in Section 4.6 we end the chapter with some conclusions.

4.1 INTRODUCTION

Nowadays, numerous platforms on the Web, such as e-commerce sites and online social networks, collect user feedback for items of several types or from multiple domains. In these cases, rather than exploiting user preference data from each domain independently, cross-domain recommender systems aim to exploit more exhaustive, multi-domain user models that allow generating item recommendations spanning several domains. As explained in Chapter 3, making use of additional knowledge from related auxiliary domains could help improve the quality of item recommendations in a target domain, e.g., by addressing the cold start problem.

These benefits rely on the assumption that there are similarities or relations between user preferences and/or item attributes from different domains. When such correspondences exist, one way to exploit them is by *aggregating knowledge* from the involved data sources (Section 3.4), for example by combining single-domain recommendations. An alternative way consists of *transferring knowledge* from a source domain to a target domain (Section 3.5), for example by sharing implicit latent features that relate source and target domains, and by exploiting implicit rating patterns from source domains in the target domain. In either of the above cases, most of the existing approaches to cross-domain recommendation are based on collaborative filtering, since it merely needs rating data, and does not require information about the users' and items' characteristics, which are usually highly heterogeneous among domains.

Social tags as inter-domain content features

Inter-domain links established through content-based features and relations, however, may have several advantages, such as a better interpretability of the cross-domain user models and recommendations, and the establishment of more reliable methods to support the knowledge transfer between domains. In particular, social tags assigned to different types of items –such as movies, music, and books– may act as a common vocabulary between domains (Enrich et al., 2013; Shi et al., 2011). Hence, as domain independent content-based features, tags can be used to overcome the information heterogeneity across domains, and are suitable for building the above mentioned interdomain links.

Integrating social tag and rating latent factors

In this chapter, we review state of the art cross-domain recommendation approaches that utilize social tags to exploit knowledge from an auxiliary domain for enhancing collaborative filtering rating predictions in a target domain. Specifically, we focus on several extensions of the matrix factorization technique proposed in (Enrich et al., 2013), which incorporates latent factors related to the users' tags. By jointly learning tag factors in both the source and target domains, hidden correlations between ratings and tags in the source domain can be transferred and used in the target domain. Hence, for instance, a movie recommender system may estimate a higher rating for a particular movie tagged as `interesting` or `amazing` if these tags are usually assigned to books positively rated. Also, books tagged as `romantic` or `suspenseful` may be recommended to a user if it is found that such tags correlate with high movie ratings. Moreover, these methods do not require common users or items between the domains, and only need a set of shared tags to bridge the domains and transfer the knowledge, a scenario more easily found in real-world applications.

Separating social tag latent factors for users and items

Enrich et al., (2013) presented several recommendation models that exploit different sets of social tags when computing rating predictions, namely tags assigned by the active user to the item for which the rating is estimated, and all the tags assigned by the community to the target item. Despite their good performance, these models do have difficulties in cold start situations where no tagging information is available for the target user/item. To overcome these limitations, in the chapter we present a model that expands user and item profiles. More specifically, we propose to incorporate additional parameters to the above models, separating user and item latent tag factors in order to capture the contributions of each to the ratings more accurately. Furthermore, by modeling user and item tags independently we are able to compute rating predictions even when a user has not assigned any tag to an item, or for items that have not been tagged yet. For such purpose, we adapt the gSVD++ algorithm (Manzato, 2013) –designed to integrate content metadata into the matrix factorization process– for modeling social tags in the cross-domain recommendation scenario.

Through a series of experiments in the movies and books recommendation domains, we show that the proposed approach outperforms state of the art methods, and validate a main contribution of this work: a novel model that separately captures user and item tagging information, and effectively transfers auxiliary knowledge to the target domain in order to provide useful cross-domain recommendations for cold start users.

4.2 BACKGROUND AND RELATED WORK

In this section we review related work on recommender systems that exploit social tags as user preferences, first focusing on single-domain systems, and then on cross-domain systems that use tags as a knowledge bridge between domains.

Social tagging systems (Marinho et al., 2011) became some of the most popular applications in the era of the so-called Social Web (or Web 2.0) to exchange and classify user generated content. In these systems, users are able to upload items, and annotate them with freely chosen words, referred as *tags*. Delicious[1], Last.fm[2] and Flickr[3] are well known examples of social tagging systems for websites, music and photos, respectively. In this context, the set of tags can be viewed as a collaborative, unstructured scheme, commonly known as *folksonomy*. *Folksonomy*

Formally, a folksonomy is a tuple $\mathcal{F} = (\mathcal{U}, \mathcal{I}, \mathcal{T}, \mathcal{A})$, where $\mathcal{U}$ and $\mathcal{I}$ are the sets of users and items, respectively, $\mathcal{T}$ is the set of tags that comprise the vocabulary expressed by the folksonomy, and $\mathcal{A} \subseteq \mathcal{U} \times \mathcal{I} \times \mathcal{T}$ is the set of assignments (annotations) of each tag t to an item i by a user u. Throughout the chapter, we will use the notation

1 Delicious, Social bookmarking, http://delicious.com
2 Last.fm, Internet radio and music catalogue, http://www.last.fm
3 Flickr, Photo sharing, http://www.flickr.com

T_u and T_i to refer to the sets of tags assigned by user u and to item i, respectively, i.e., $T_u = \{t | (u, t, \cdot) \in \mathcal{A}\} \subseteq \mathcal{T}$ and $T_i = \{t | (\cdot, t, i) \in \mathcal{A}\} \subseteq \mathcal{T}$.

4.2.1 *Social tags as a source of user preferences*

Social tags as user preferences

From the point of view of the users of a social tagging service, folksonomies may have multiple purposes, such as organizing a collection of resources, sharing them with friends or other users, or even promoting certain resources by annotating them with popular tags. Nonetheless, users more generally use tags that reflect their tastes and interests. Golder and Huberman, (2006) analyzed tags on Delicious, and observed that (a) most of the tags are used to identify topics of the tagged resources, and (b) usually resources are annotated for personal use rather than for the benefit of the community. These observations lead to the development of personalized search and recommendation systems based on the assumption that tags represent user preferences and accurate item features.

Social tag-based personalized search

In the context of personalized search, Hotho et al., (2006) proposed *FolkRank*, an extension of the well known PageRank algorithm, which exploits the graph structure of folksonomies to support personalized ranking of users, items, and tags within the folksonomy. Later, Noll and Meinel, (2007) presented an approach for re-ranking web search results taking into account user preferences by matching tag-based user profiles and an index of documents annotated with search keywords. In the same context, Xu et al., (2008) proposed to rank web pages based not only on matching terms, but also on the similarities between the users' preferred topics and the web pages topics, which were represented by tags from certain folksonomy.

Social tag-based recommendation

Regarding the use of tags as user preferences for recommendation, Cantador et al., (2010b) analyzed and compared a number of content-based approaches based on the Vector Space Model (Baeza-Yates and Ribeiro-Neto, 1999) that exploit social tags to build user and item profiles. Sen et al., (2009) presented a two-step approach that first infers the users' preferences for tags based on their tagging activities and the tags assigned to liked items, and then generates item recommendations from the estimated preferences. More recently, Gemmell et al., (2012) proposed a linear-weighted hybrid approach that combines the predictions of traditional user-user and item-item collaborative filtering approaches, content-based estimations based on tags, item popularity scores, and pairwise factorization of the user-item-tag tensor.

4.2.2 *Social tag-based cross-domain recommendation*

Domain-independent social tags

Social tags assigned by users provide a way to represent the content of the items in a convenient manner for cross-domain recommendation. In contrast to other content-based features such as movie actors or book genres, social tags consist of short pieces of text in the form of keywords that potentially express a much broader set of item properties. In fact, in most social tagging systems, the tags do not belong to a limited, fixed vocabulary, but rather are freely chosen by the users, who annotate the items as specific or as generic as they want. For instance, users can annotate a movie with specific tags like `film noir` and `oscar winning`, and with more general tags like `exciting` and `masterpiece`. In this context, it is important to note that more generic tags may not be exclusive to a particular domain, and can be applied to other domains –e.g., a book can also be annotated as `exciting` or `masterpiece`–, thus providing a subset of domain-independent features suitable for representing items with different sets of attributes. This observation makes social tags particularly appealing for cross-domain recommendation scenarios, in which the subset of shared tags can be used to establish a bridge between the involved domains, allowing the transfer of knowledge. As a consequence, most approaches that exploit social tags do not require overlap of users or items to bridge the domains, relying exclusively on shared tags (see Table 3.6 and Table 3.7).

Social tag matrix factorization

In Section 3.4 we reviewed representative works that merge, mediate and combine user profiles from social tagging systems in order to build a unified profile that captures user preferences across domains. In the following we focus on knowledge linkage and transfer approaches that exploit social tags within the matrix factorization framework, as they form the basis of our approach. In particular, we consider and describe the models proposed by Shi et al., (2011), and Enrich et al., (2013) as the best representative approaches of explicit domain linking (Section 3.5.1) and implicit sharing of latent features (Section 3.5.2), respectively. It is worth noting that both approaches, as most proposed so far for cross-domain recommendation, address the rating prediction task as opposed to the item ranking task (Section 2.1.1).

Social tag-based inter-domain similarities

The model presented in (Shi et al., 2011) exploits social tags shared between the domains to compute user-user and item-item inter-domain similarities that link the domains. Specifically, the authors define a matrix $\mathbf{S}^{\text{user}}$ that captures similarities between the users in the source domain $\mathcal{U}_S$ and the users in the target domain $\mathcal{U}_T$, so that the similarity between $u \in \mathcal{U}_S$ and $v \in \mathcal{U}_T$ is computed as:

$$\mathbf{S}^{\text{user}}_{uv} = \frac{|T_u \cap T_v|}{|T_u||T_v|} \tag{4.1}$$

Likewise, the item-item similarity matrix $\mathbf{S}^{\text{item}}$ captures the tag-based similarity between item i in the source domain $i \in \mathcal{I}_S$ and item j in the target $j \in \mathcal{I}_T$:

$$\mathbf{S}^{\text{item}}_{ij} = \frac{|T_i \cap T_j|}{|T_i||T_j|} \tag{4.2}$$

Joint factorization of social tag-based similarity and rating matrices

Then, the inter-domain similarity matrices are jointly factorized with the rating matrices in both domains in a collective fashion, so that user and item latent vectors not only reconstruct the ratings, but also the computed tag-based similarities. This is reflected in the following loss function, where the rating matrices of the source $\mathbf{R}_S$ and target $\mathbf{R}_T$ are simultaneously factorized:

$$\begin{aligned}
&\mathcal{L}(\mathbf{P}^S, \mathbf{Q}^S, \mathbf{P}^T, \mathbf{Q}^T) = \\
&\sum_{(u,i)\in\mathcal{R}_S} \left(r^S_{ui} - \langle \vec{p}^S_u, \vec{q}^S_i \rangle\right)^2 + \sum_{(v,j)\in\mathcal{R}_T} \left(r^T_{vj} - \langle \vec{p}^T_v, \vec{q}^T_j \rangle\right)^2 + \\
&\alpha \sum_{u\in\mathcal{U}_S} \sum_{v\in\mathcal{U}_T} \left(\mathbf{S}^{\text{user}}_{uv} - \langle \vec{p}^S_u, \vec{p}^T_v \rangle\right)^2 + \beta \sum_{i\in\mathcal{I}_S} \sum_{j\in\mathcal{I}_T} \left(\mathbf{S}^{\text{item}}_{ij} - \langle \vec{q}^S_i, \vec{q}^T_j \rangle\right)^2 + \\
&\lambda \left(\|\mathbf{P}^S\|^2 + \|\mathbf{P}^T\|^2 + \|\mathbf{Q}^S\|^2 + \|\mathbf{Q}^T\|^2 \right)
\end{aligned} \tag{4.3}$$

The parameters α and β control the contribution of the inter-domain similarities in the learning of the latent feature vectors. As usual in matrix factorization models for rating prediction, the parameters $\mathbf{P}^S, \mathbf{Q}^S, \mathbf{P}^T, \mathbf{Q}^T$ are automatically learned by minimizing Equation 4.3 using Stochastic Gradient Descent.

Incorporating social tag latent factors into the rating prediction

Rather than linking the domains with tag-based similarities, Enrich et al., (2013) included additional latent vectors for the tags into the rating prediction function, which are in turn shared between the domains. The authors proposed three models, all of which are based on the assumption that the effect of tags on the ratings can be reused across domains. For instance, if a tag such as `amazing` is often assigned with high ratings in the source domain, then this correlation can be transferred and exploited in the target domain to generate more accurate rating predictions. This implicit dependencies are captured by introducing a new set of latent vectors for the tags, $\vec{y}_t \in \mathbb{R}^k$, where k is the dimension of the latent space, same as for users and items as in standard matrix factorization.

The UserItemTags model

The first model, *UserItemTags*, exploits the set $T_u(i)$ of tags assigned by the active user u to the target item i. Hence, this model assumes that the user already tagged the item, even if she did not rate it yet. Ratings in both domains are then estimated as:

$$\hat{r}(u,i) = \left\langle \vec{p}_u, \vec{q}_i + \frac{1}{|T_u(i)|} \sum_{t\in T_u(i)} \vec{y}_t \right\rangle \tag{4.4}$$

We note here that if the user has not tagged the item, i.e., $T_u(i) = \emptyset$, then the tag component does not play any role, and the model behaves exactly as standard matrix factorization. Also, even though the tag factors $\vec{y}_t$ are only combined with the item factors $\vec{q}_i$, the user and item factorization components are not completely uncoupled in Equation 4.4, since the set $T_u(i)$ still depends on user u.

The UserItemRelTags model

An improvement over the previous model was also presented in (Enrich et al., 2013), based on the observation that not all the tags are equally relevant (i.e., discriminative) to predict the ratings. The proposed alternative is to filter from the set $T_u(i)$ those tags that are not relevant according to certain criterion. In particular, the authors used Wilcoxon's rank-sum test for each tag to decide if the mean rating significantly changes in the presence/absence of the tag in the dataset. In the model rating predictions are computed in an analogous manner as before:

$$\hat{r}(u,i) = \left\langle \vec{p}_u, \vec{q}_i + \frac{1}{|TR_u(i)|} \sum_{t \in TR_u(i)} \vec{y}_t \right\rangle \tag{4.5}$$

where the set $TR_u(i) \subseteq T_u(i)$ only contains those tags for which the p-value of the obove mentioned test is $p < 0.05$. Hereafter, we will refer to this method as *UserItemRelTags*.

The ItemRelTags model

As noted by the authors, the previous methods are useful when the user has tagged, but not rated the item. These methods, however, do not greatly improve over standard matrix factorization in cold start situations where new users or items are considered. Aiming to address this limitation, they proposed a third approach called the *ItemRelTags* model:

$$\hat{r}(u,i) = \left\langle \vec{p}_u, \vec{q}_i + \frac{1}{|TR(i)|} \sum_{t \in TR(i)} \vec{y}_t \right\rangle \tag{4.6}$$

In this case, the set $TR(i)$ contains all the relevant tags assigned by the whole community to the item i, with possible repetitions, so that tags that appear more often contribute with more factors. Being n_it the number of times tag t was assigned to item i, the normalization factor is defined as $|TR(i)| = \sum_{t \in TR(i)} n_it$.

We note that the set $TR(i)$ does not depend on the target user u, and that the user and item components of the factorization in Equation 4.6 are fully uncoupled. This has the advantage that tag factors can also be exploited in the rating predictions for users with no tagging information, outperforming the standard matrix factorization. The *ItemRelTags* model, however, does not take into account the possibility that the user has tagged items different than the one for which the rating is being estimated. In such a case, it may be beneficial to enrich the user's profile by considering other tags the user has chosen in the past as evidence of her preferences. In the next section, we propose a model that aims to exploit this information to generate more accurate recommendations.

Joint factorization of tag-based rating matrix

In the three previous models, the rating matrices of the source and target domains are jointly factorized in a collective manner, making no distinction between the source and target domain ratings. When there is no user and item overlap, the transfer of knowledge is only supported by the factors $\vec{y}_t$ associated to tags shared between the domains, i.e., $t \in \mathcal{T}_s \cap \mathcal{T}_T$. Similarly to standard matrix factorization, the model parameters are learned by minimizing the associated squared loss function:

$$\mathcal{L}(\mathbf{P}^S, \mathbf{Q}^S, \mathbf{P}^T, \mathbf{Q}^T, \mathbf{Y}) = \sum_{(u,i) \in \mathcal{R}_S} \left(r^S_{ui} - \hat{r}(u,i)\right)^2 + \sum_{(v,j) \in \mathcal{R}_T} \left(r^T_{vj} - \hat{r}(v,j)\right)^2 + \lambda \left(\|\mathbf{P}^S\|^2 + \|\mathbf{P}^T\|^2 + \|\mathbf{Q}^S\|^2 + \|\mathbf{Q}^T\|^2 + \|\mathbf{Y}^T\|^2 \right) \quad (4.7)$$

where the matrix $\mathbf{Y}$ contains all the $\vec{y}_t$ as rows, and the rating predictions $\hat{r}(\cdot, \cdot)$ are respectively computed using Equations (4.4),(4.5),(4.6) for each model. We note that in the previous loss function only the tag factors $\vec{y}_t$ contribute to both the predictions of source and target ratings through the terms $\hat{r}(u,i)$ and $\hat{r}(v,j)$, respectively.

4.3 PROPOSED RECOMMENDATION MODEL

In this section we present *TagGSVD++*, our social tag-based model for cross-domain recommendation, also focusing on the rating prediction task. The model is built upon the approach proposed by Enrich et al., (2013), by enhancing the factorization scheme to address limitations of such approach. In particular, we base our model on the gSVD++ algorithm by Manzato, (2013), which extends the standard matrix factorization model to incorporate implicit user feedback and item metadata into the rating predictions. Hence, we first briefly describe gSVD++ before presenting our model.

4.3.1 *Adding item metadata to matrix factorization*

The gSVD++ algorithm

The gSVD++ algorithm (Manzato, 2013) extends the popular SVD++ method (Koren, 2008) considering information about the items attributes in addition to the users' implicit feedback.

The model introduces a new set of latent variables $\vec{x}_g \in \mathbb{R}^k$ for metadata that complement the item factors. This idea combined with the SVD++ algorithm leads to the following formula for computing rating predictions:

$$\hat{r}(u,i) = \left\langle \vec{p}_u + \frac{1}{\sqrt{|N(u)|}} \sum_{j \in N(u)} \vec{y}_j, \vec{q}_i + \frac{1}{|G(i)|^{\beta}} \sum_{g \in G(i)} \vec{x}_g \right\rangle \quad (4.8)$$

where $N(u)$ is the set of user u's implicit feedback, i.e., the set of items rated by user u. The set $G(i)$ contains the attributes related to item i, e.g., *comedy* and *romance* in the case of movie genres. The parameter β is set to 1 when $G(i) \neq \emptyset$ and 0 otherwise. We note that in the previous formula, both user and item factors are enriched with new uncoupled latent variables that separately capture information about the users and items, leading to a symmetric model with four types of parameters. Again, parameter learning can be performed by minimizing the associated squared error function through gradient descent:

$$\mathcal{L}(\mathbf{P},\mathbf{Q},\mathbf{X},\mathbf{Y}) = \sum_{(u,i)\in\mathcal{R}} (r_{ui} - \hat{r}(u,i))^2 + \lambda\left(\|\vec{p}_u\|^2 + \|\vec{q}_i\|^2 + \sum_{g\in G(i)} \|\vec{x}_g\|^2 + \sum_{j\in N(u)} \|\vec{y}_j\|^2\right) \tag{4.9}$$

The use of additional latent factors for item metadata is reported to improve prediction accuracy over SVD++ in (Manzato, 2013). In the next subsection we adapt this model to separately learn user and item tag factors, aiming to support the transfer of knowledge between domains.

4.3.2 *The TagGSVD++ model*

Although the recommendation models presented in (Enrich et al., 2013) are capable of transferring tagging information between domains, they suffer from some limitations. The *UserItemTags* and *UserItemRelTags* models cannot do better than the standard matrix factorization if the user has not tagged the item for which the rating is being estimated, while the *ItemRelTags* model does not fully exploit the user's preferences expressed in the tags assigned to other items.

Adapting the gSVD++ algorithm

Here we propose to adapt the gSVD++ algorithm by introducing an additional set of latent variables $\vec{x}_s \in \mathbb{R}^k$ that enrich the user's factors, and better capture the effect of the user's tags on the rating estimation process. Specifically, we distinguish between two different sets of tags for users and items, and factorize the rating matrix into fully uncoupled user and item components as follows:

$$\hat{r}(u,i) = \left\langle \vec{p}_u + \frac{1}{|T_u|}\sum_{s\in T_u} n_{us}\vec{x}_s, \vec{q}_i + \frac{1}{|T_i|}\sum_{t\in T_i} n_{it}\vec{y}_t \right\rangle \tag{4.10}$$

where the set T_u contains all the tags assigned by user u to any item. Respectively, T_i is the set of tags assigned by any user to item i, and plays the role of item metadata $G(i)$ in the gSVD++ algorithm (see Equation 4.8). As in the *ItemRelTags* model, there may be repeated tags in each of the above tag sets, which we account for by considering the number of times a tag appears in T_u or T_i. In Equation 4.10, n_{us} is the number of items on which the user u applied tag s, and n_{it} is the number of users that applied tag t to item i. As previously, tag factors are normalized by $|T_u| = \sum_{s \in T_u} n_{us}$ and $|T_i| = \sum_{t \in T_i} n_{it}$, so that factors $\vec{x}$ and $\vec{y}_t$ do not dominate over the rating factors $\vec{p}_u$ and $\vec{q}_i$ for users and items with a large number of tags.

In the proposed model, which we call *TagGSVD++*, a user's profile is enhanced with the tags the user utilized, hypothesizing that her preferences are better captured, and that transferring tag information between domains is beneficial for estimating ratings in the target domain. Likewise, item profiles are extended with the tags that were assigned to them, as in the *ItemRelTags* model.

We remark that TagGSVD++, as well as the models proposed in (Enrich et al., 2013), is a collective approach, where the rating matrices of the source and target domains are simultaneously factorized. Moreover, the transfer of knowledge is symmetric since the model does not distinguish between source and target domains. Hence, assuming there is no user or item overlap between the domains, the joint factorization of the source and target matrices is equivalent to the factorization of the single matrix that results from concatenating the rating matrices of both domains, so that $\mathcal{R} = \mathcal{R}_S \cup \mathcal{R}_T$. We make use of this observation in order to simplify the notation in the derivations that follow, also noting that now $u \in \mathcal{U}_S \cup \mathcal{U}_T$ and $i \in \mathcal{I}_S \cup \mathcal{I}_T$.

The parameters of *TagGSVD++* are automatically learned from the observed training data by minimizing the corresponding regularized squared loss function:

$$
\begin{aligned}
\mathcal{L}(\mathbf{P}, \mathbf{Q}, \mathbf{X}, \mathbf{Y}) &= \sum_{(u,i) \in \mathcal{R}} \frac{1}{2} \ell(\vec{p}_u, \vec{q}_i, \{\vec{x}_s\}_{s \in T_u}, \{\vec{y}_t\}_{t \in T_i}) \\
&= \sum_{(u,i) \in \mathcal{R}} \frac{1}{2} \left(r_{ui} - \left\langle \vec{p}_u + \frac{1}{|T_u|} \sum_{s \in T_u} n_{us} \vec{x}_s, \vec{q}_i + \frac{1}{|T_i|} \sum_{t \in T_i} n_{it} \vec{y}_t \right\rangle \right)^2 \\
&\quad + \frac{\lambda}{2} \left(\|\vec{p}_u\|^2 + \|\vec{q}_i\|^2 + \sum_{s \in T_u} \|\vec{x}_s\|^2 + \sum_{t \in T_i} \|\vec{y}_t\|^2 \right)
\end{aligned}
\tag{4.11}
$$

Based on the previous observation, we have removed the explicit distinction between the source and target domains, treating all the ratings as if they belong to a single set $\mathcal{R}$. Assuming that neither the users nor items overlap, each user-item pair $(u, i) \in \mathcal{R}$ can be univocally identified either in $\mathcal{R}_S$ or in $\mathcal{R}_T$.

Stochastic Gradient Descent in the TagGSVD++ model

We use the *Stochastic Gradient Descent* algorithm to find a local minimum of the function $\mathcal{L}$ by iteratively updating the parameters after each observed $(u, i) \in \mathcal{R}$ pair. In general, if θ is a model parameter, stochastic gradient descent works by shifting θ in the direction of maximum descent of the local loss ℓ, given by its gradient:

$$\theta \leftarrow \theta - \eta \frac{\partial \ell}{\partial \theta} \tag{4.12}$$

Algorithm 2 Stochastic gradient descent for *TagGSVD++*

```
procedure TRAIN
    Initialize the parameters at random
    for epoch ← 1, ..., N do
        SHUFFLE(R)
        for all (u, i) ∈ R do
            p_u ← p_u − η ∂ℓ/∂p_u                      ▷ Using Equation 4.13
            q_i ← q_i − η ∂ℓ/∂q_i                      ▷ Using Equation 4.14
            x_a ← x_a − η ∂ℓ/∂x_a    ∀a ∈ T_u          ▷ Using Equation 4.15
            y_b ← y_b − η ∂ℓ/∂y_b    ∀b ∈ T_i          ▷ Using Equation 4.16
        end for
    end for
end procedure
```

where η is the *learning rate* that determines to what extent the parameter is updated in each iteration. A small learning rate can make the learning slow, whereas a large learning rate may make the algorithm fail to converge. For the sake of completeness, we provide the derivatives of the loss function needed for stochastic gradient descent with respect to the parameters in our model:

$$\frac{\partial \ell}{\partial \vec{p}_u} = -e_{ui}\left(\vec{q}_i + \frac{1}{|T_i|}\sum_{t\in T_i} n_{it}\vec{y}_t\right) + \lambda\vec{p}_u \tag{4.13}$$

$$\frac{\partial \ell}{\partial \vec{q}_i} = -e_{ui}\left(\vec{p}_u + \frac{1}{|T_u|}\sum_{s\in T_u} n_{us}\vec{x}_s\right) + \lambda\vec{q}_i \tag{4.14}$$

$$\frac{\partial \ell}{\partial \vec{x}_a} = -e_{ui}\frac{n_{ua}}{|T_u|}\left(\vec{q}_i + \frac{1}{|T_i|}\sum_{t\in T_i} n_{it}\vec{y}_t\right) + \lambda\vec{x}_a \quad \forall a \in T_u \tag{4.15}$$

$$\frac{\partial \ell}{\partial \vec{y}_b} = -e_{ui}\frac{n_{ib}}{|T_i|}\left(\vec{p}_u + \frac{1}{|T_u|}\sum_{s\in T_u} n_{us}\vec{x}_s\right) + \lambda\vec{y}_b \quad \forall b \in T_i \tag{4.16}$$

where $e_{ui} \triangleq r_{ui} - \hat{r}(u,i)$ is the error in the prediction of rating r_{ui}.

The previous gradients are combined with the update rule in Equation 4.12, and are applied simultaneously for all the parameters after each (u, i) pair. In offline settings, as in our experiments, this process is typically repeated a fixed number of times N or *epochs* over the whole training set. We summarize all the steps needed to train *TagGSVD++* in Algorithm 2.

4.4 EXPERIMENTS

We have evaluated the proposed *TagGSVD++* model in a cross-domain collaborative filtering setting, by empirically comparing it against single-domain matrix factorization methods, and the state of the art cross-domain recommendation models described in Section 4.2.2, i.e., those proposed in (Enrich et al., 2013).

4.4.1 *Dataset*

We have attempted to reproduce the cross-domain dataset used in (Enrich et al., 2013), aiming to compare the results of our model with those presented in that paper. For the sake of completeness, we also describe the data collection process here.

In order to simulate the cross-domain collaborative filtering setting, we downloaded two well known, publicly available datasets for the movies and books recommendation domains, namely the MovieLens and the LibraryThing datasets. The MovieLens 10M dataset[4] (ML) contains over 10 million ratings and 100,000 tag assignments by 71,567 users to 10,681 movies. The LibraryThing dataset[5] (LT) contains over 700,000 ratings and 2 million tag assignments by 7,279 users on 37,232 books. Ratings in both datasets are expressed on a 1-5 scale, with interval steps of 0.5.

Since we were interested in analyzing the effect of tags on rating prediction, in the MovieLens dataset we only kept ratings on movies for which at least one tag was applied, leaving a total of 24,564 ratings. Also following the setup done by Enrich et al., (2013), we considered the same amount of ratings in LibraryThing, and took the first 24,564 ratings. We note, however, that the original dataset contained duplicate rows and inconsistencies, i.e., some user-item pairs had more than one rating. Hence, we preprocessed the dataset removing such repetitions and keeping only the repeated ratings that appeared first in the dataset file. We also converted the tags into lower case in both datasets. Table 4.1 shows the characteristics of the final datasets.

4.4.2 *Evaluated approaches*

As mentioned above, we compared the performance of the proposed model against single-domain baselines and the state of the art tag-based models described in Section 4.2.2. All these recommendation approaches are summarized next:

4 The MovieLens datasets, `http://grouplens.org/datasets/movielens`
5 The LibraryThing dataset, `http://www.macle.nl/tud/LT`

Table 4.1: Details of the used datasets after preprocessing.

	MovieLens	**LibraryThing**
Users	2026	244
Items	5088	12 801
Ratings	24 564	24 564
Tags	9529	4598
Tag assignments	44 805	72 943
Average ratings per user	12.12	100.67
Rating sparsity (%)	99.76	99.21
Avg. tag assignments per user	22.16	298.95
Ratio of shared tags (%)	13.81	28.62

- **MF**. The standard matrix factorization method trained by stochastic gradient descent over the observed ratings from both the movies and books datasets.
- **SVD++**. An adaptation of MF by Koren, (2008) that takes implicit data into account. In our experiments, the set $N(u)$ contained all the items rated by user u.
- **gSVD++**. An extension of SVD++ that incorporates item metadata into the factorization process. In our experiments, we have considered as set of item attributed $G(i)$ the tags T_i assigned to item i by any user. Note that, as tags are content features for both movies and books, this method is suitable for cross-domain recommendation, since knowledge can be transferred through the metadata (tag) factors. This differs from the proposed *TagGSVD++* in that users are modeled as in SVD++ by considering rated items as implicit feedback instead of their tags. Also, the normalization of the implicit data factors of the user component involves a square root; see Equation 4.8 and Equation 4.10.
- **UserItemTags**. A model that expands the profile of an item i with latent factors of tags that the target user assigned to i. Its parameters are learned by simultaneously factorizing the rating matrices of both source and target domains.
- **UserItemRelTags**. A variation of the previous model that only takes relevant tags into account, as determined by a Wilcoxon rank-sum test.
- **ItemRelTags**. Instead of tags assigned by the user, this model exploits all relevant tags applied by the whole user community,

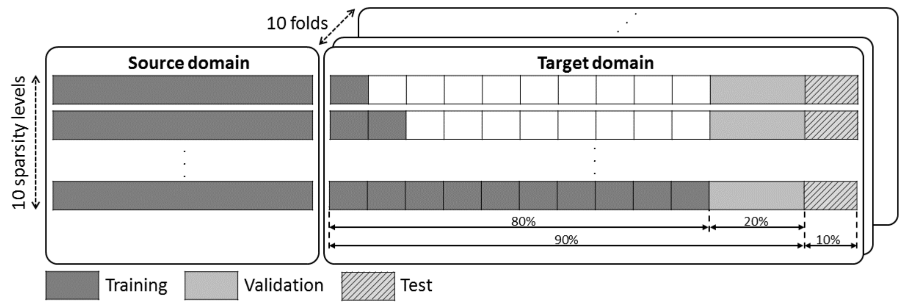

Figure 4.1: Data splitting for cross-validation with different sparsity levels. Training data consists of source domain ratings and portions of the target domain, marked in dark.

and is thus able to compute rating predictions even if the user has not tagged the target item.

4.4.3 *Evaluation methodology and metrics*

Data split

We evaluated the previous recommendation methods in two settings, using MovieLens as source domain and LibraryThing as target domain, and vice versa. In both cases, we evaluated the approaches through 10-fold cross-validation, i.e., we shuffled the target domain ratings, and split them into 10 non-overlapping folds. In each fold we left out one part, 10% of the ratings, as a test set to estimate the performance of the recommendation approaches. The remaining 90% of the ratings were used as a training set to learn the models, and a validation set to find the optimal values of the models parameters. Specifically, we randomly chose 80% of these remaining ratings, and combined them with the source domain ratings to build the models. The final 20% left was used for the validation set to select the best number of factors k, learning rate η, and regularization parameter λ. Figure 4.1 depicts the split of the data into training, validation and test sets.

As in (Enrich et al., 2013), we also wanted to investigate how the number of available ratings in the target domain affects the quality of generated recommendations. For such purpose, we further split the training data from the target domain into 10 portions to simulate different rating sparsity levels. First, in order to evaluate the performance of the approaches in cold start situations, we used only 10% of the target ratings, i.e., $0.1 \times 0.8 \times 0.9 \times 24,564 = 1,768$ ratings (see Table 4.1). Then, we incrementally added additional 10% of the ratings to analyze the behavior of the approaches with an increasingly larger amount of observed rating data. In each sparsity level, the full set of source domain ratings was also used to build the models.

Evaluation metric

Since all the approaches were designed for the rating prediction task, we measured their performance as the accuracy of the estimated ratings. Moreover, aiming to compare our results against those re-

Table 4.2: Average values for the best parameters obtained.

	ML → LT			LT → ML		
	k	η	λ	k	η	λ
MF	41	0.020	0.009	43	0.020	0.009
SVD++	41	0.020	0.007	43	0.020	0.006
gSVD++	43	0.019	0.001	43	0.020	0.004
UserItemTags	46	0.019	0.003	46	0.020	0.010
UserItemRelTags	39	0.017	0.008	41	0.020	0.017
ItemRelTags	40	0.017	0.001	46	0.020	0.006
TagGSVD++	40	0.013	0.036	46	0.019	0.045

ported in (Enrich et al., 2013), we compute the Mean Absolute Error (MAE) of each model in the different settings described above as in that paper:

$$\mathrm{MAE} = \frac{1}{|\mathcal{R}_{\text{test}}|} \sum_{(u,i) \in \mathcal{R}_{\text{test}}} |r_{ui} - \hat{r}(u,i)| \tag{4.17}$$

where $\mathcal{R}_{\text{test}}$ contains the ratings in the test set that we left out for evaluation.

4.5 RESULTS

As previously mentioned, we reserved 20% of the target domain training data in each fold for validating the models and finding their best parameters, in order not to overestimate the performance results.

Parameter setting

For hyperparameter optimization, with each model and sparsity level in the target domain, we performed a grid (stepsize) search on the validation set for the values of the learning rate η, the amount of regularization λ, and the number of latent features k. To get an idea of the typical values obtained for the parameters, Table 4.2 shows the average best values for each approach. From the table, we observe that there is not a large difference in the optimal number of factors and learning rates between configurations. In contrast, we note that the amount of regularization needed for the proposed *TagGSVD++* model is relatively large, e.g., comparing $\lambda = 0.036$ of *TagGSVD++* with $\lambda = 0.009$ of MF. This may be due to the additional set of latent variables for tags that our model uses; more complex models are able to account for greater variance in the data and tend to overfit more easily, thus requiring more regularization. In order to analyze how the available information in the target domain affects the stability of the model, Figure 4.2 shows the optimal value for the regularization parameter for several sparsity levels.

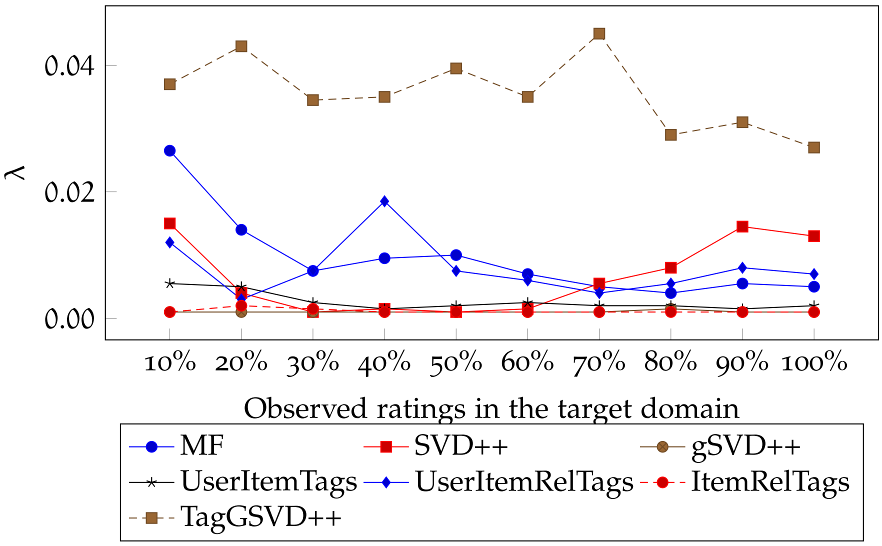

Figure 4.2: Optimal values for the regularization parameter using MovieLens as source domain and LibraryThing as target domain.

We note that the gSVD++ algorithm, upon which our model is built, also introduces additional latent variables, and yet requires a lower regularization. We argue that the differences between gSVD++ and TagGSVD++ regularizations are caused by the $N(u)$ and T_u sets; see equations (4.8) with $G(i) = T_i$ and (4.10). In Table 4.1 we see that, on average, the number of tags assigned by a user is much larger than the number of rated items. This results in more variables that are actually taking part in the rating predictions, and hence in a more complex model that requires more regularization to prevent overfitting.

Once we obtained the best parameters for each sparsity level, we ran the approaches separately on the test set of each fold. The final performance was estimated as the average MAE across the 10 folds. Figure 4.3 shows the results obtained using LibraryThing as source domain and MovieLens as target domain. All the differences with respect to our TagGSVD++ model are statistically significant as determined with a Wilcoxon signed rank test at the 95% confidence level. It can be seen that the proposed TagGSVD++ model is able to consistently outperform the other approaches for all sparsity levels in the target domain, also in the cold start when only 10%–20% of the ratings are available. We also note that cross-domain methods always achieve better accuracy than single-domain MF, although SVD++ effectively exploits implicit feedback and remains competitive until the 50% sparsity level. Then, as the sparsity decreases, cross-domain models provide greater improvements. This indicates that even if plenty of the target domain rating data is available, it is still beneficial to transfer knowledge from the source domain.

Results in the movies $\rightarrow$ books *configuration*

The results using MovieLens as source domain and LibraryThing as target domain, i.e., in the movies $\rightarrow$ books configuration, are

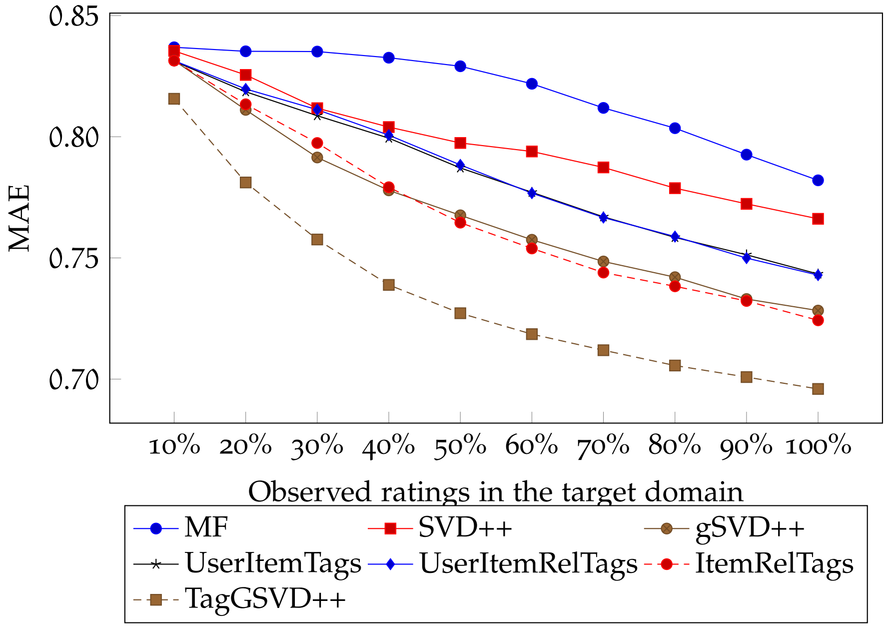

Figure 4.3: Average MAE over the 10 folds using LibraryThing as source domain and MovieLens as target domain.

shown in Figure 4.4. As before, the difference in MAE between TagGSVD++ and the other approaches is statistically significant, according to the Wilcoxon signed rank test with 95% confidence level. Again, TagGSVD++ is the best performing approach for all rating sparsity level, followed by the cross-domain methods. We now observe that the values of MAE are in general larger than in the movies case, which seems to indicate that the transfer of knowledge is not as effective in this setting. This observation is in accordance with the results reported in (Enrich et al., 2013), where the authors argue that this may be caused by differences in the ratio of overlapping tags between the domains. Only 13.81% of the tags in MovieLens are shared in LibraryThing (see Table 4.1), and thus less latent tag factors learned in the source domain can be used in the target to compute rating predictions.

In order to understand the performance of the models for cold start users, we also analyze their behavior for users with different amounts of observed feedback in the target domain. Figure 4.5 shows the average rating prediction error for groups of users in terms of the number of available ratings and tag assignments, both using LibraryThing as source and MovieLens as target (left) and vice versa (right). The plots on the right do not contain values in the ranges $(0, 5]$ and $(5 - 10]$ because in our LibraryThing dataset, each user has at least 11 ratings, hence the different x axis with respect to the left figures.

Our TagGSVD++ model achieves the best performance in almost every configuration. Moreover, its improvement over the baselines is greater in cold start situations, where a user only rated or tagged

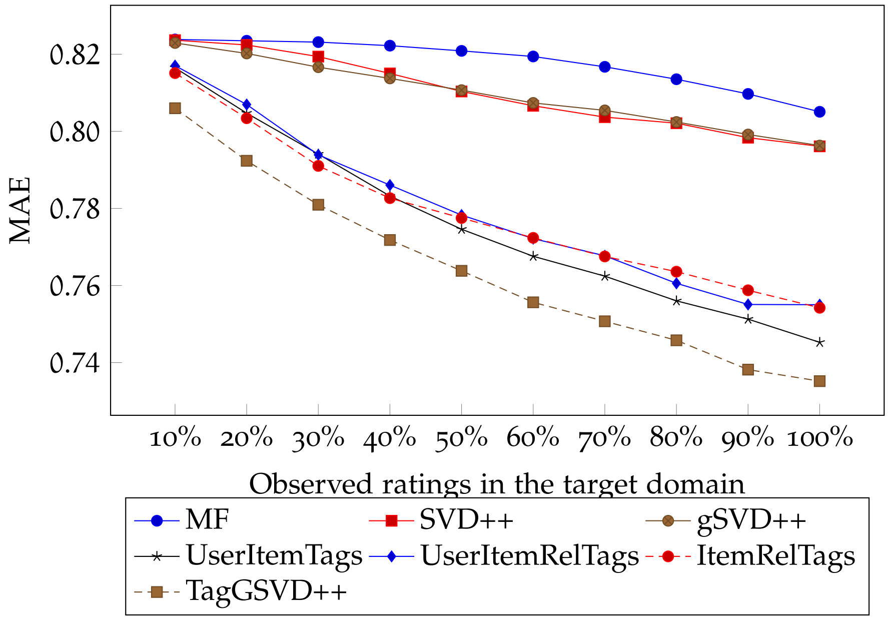

Figure 4.4: Average MAE over the 10 folds using MovieLens as source domain and LibraryThing as target domain.

a few target domain items, indicating that the additional factors for user tags, and the uncoupling of the user and item terms allow for a more effective knowledge transfer in such scenario. In the books → movies configuration, we observe that as more feedback is available all the methods provide more accurate predictions. However, it is remarkable that once $(20-50]$ ratings are observed, the performance of cross-domain methods (all but MF and SVD++) starts degrading while single-domain methods stabilize. This indicates a *negative transfer* phenomenon (Pan and Yang, 2010): the amount of preferences in the target domain is now enough to generate accurate predictions, and transferring cross-domain information is no longer useful. Finally, we note that all the methods struggle when the users tagged/rated more than 100 items. We argue that even though more training data is available for the algorithms in this case, they are not able to generalize to unseen data, likely as a symptom of an overfitting problem. We claim, nonetheless, that this is no longer a cold start scenario, and is thus out of the scope of this thesis. Our focus here is in addressing cold start situations; other types of algorithms may be more suitable for general cases with a large number of ratings per user.

Results in the books → movies configuration

The above observations also apply in the ML → LT configuration, but we note the anomalous behavior when the users tagged less than 20 books. Upon further inspection, we observed that only two users were in this group, and that there is no cold start effect regarding tags in the LibraryThing dataset. Moreover, 95% of the users had tagged

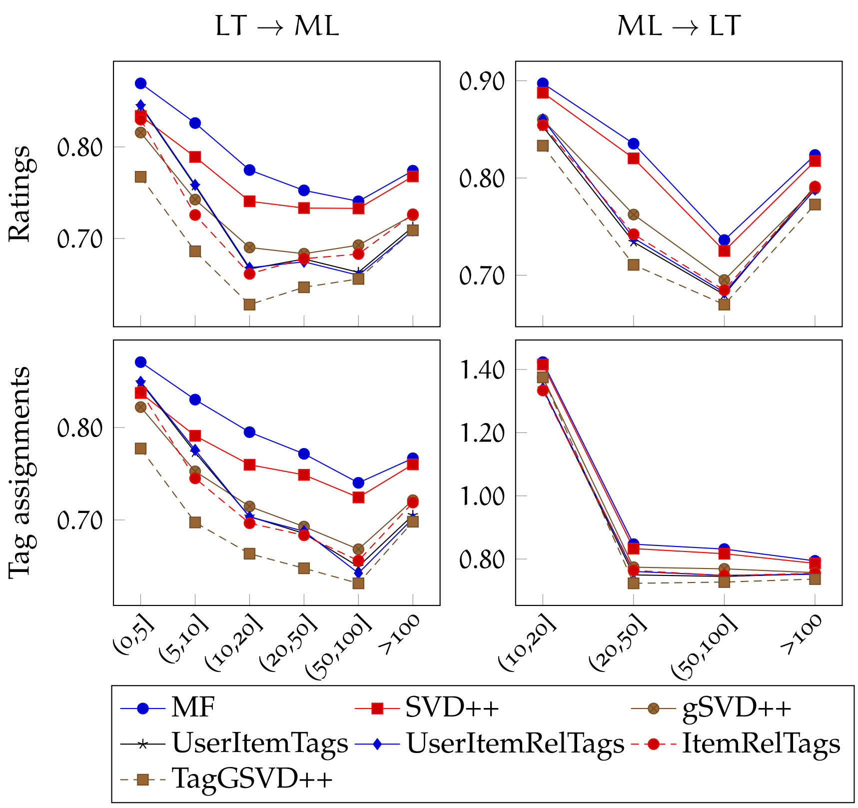

Figure 4.5: Average MAE for users with different amounts of observed ratings and tag assignments in the target domain. On the left figures, LibraryThing is used as source and MovieLens as target. Right, MovieLens as source and LibraryThing as target.

at least 28 items, and the average number of tags per user is nearly 300 (see Table 4.1).

4.6 CONCLUSIONS

One of the major difficulties that arises in cross-domain recommendation is how to link or relate the different domains to support the transfer of knowledge. Due to the common heterogeneity of item content-based features across domains, collaborative filtering techniques have become more popular than content-based methods. However, recent work (Enrich et al., 2013; Shi et al., 2011) has concluded that more reliable and meaningful relations can be established between the domains by exploiting social tags.

In this chapter, we have adapted a novel extension of the well known SVD++ algorithm (Koren, 2008) to separately model the effect of user and item tags on the observed ratings. By introducing a new set of latent variables that represent tags in the user profiles, our TagGSVD++ model is able to transfer knowledge from a source domain effectively, providing more accurate rating predictions in the target domain, especially in cold start situations. From our experiments on the movies and books recommendation domains, we conclude that exploiting additional tag factors, and decoupling user and item components in the factorization process improve the transfer of knowledge, and the accuracy of recommendations.

Regarding the categorization of cross-domain recommendation approaches proposed in Chapter 3, the tag-based matrix factorization model presented in this chapter belongs to the category of **knowledge transfer** approaches, and is aimed to address the **linked-domain exploitation** cross-domain task with the goals of addressing the user cold start problem and improving the overall accuracy of the system. For comparison purposes with the state of the art, our model also targets the rating prediction task with explicit user feedback, which we evaluated using an error-based metric as in (Enrich et al., 2013). However, as we mentioned in Section 2.1.1, the item ranking task is arguably more realistic, and thus may not be seen as a limitation of the approaches described in this chapter.

5 EXPLOITING USER PERSONALITY FACTORS IN MATRIX FACTORIZATION FOR CROSS-DOMAIN COLLABORATIVE FILTERING

Personality is a pattern of values, attitudes, thoughts, feelings and behavioral repertoire —habits, skills and social relationships— that characterizes a person, and has certain persistence and stability over the life, so that the manifestations of that pattern in different situations has some degree of predictability. In fact, in some domains, people with similar personality traits tend to have similar preferences, which make personality a potential source of information to provide collaborative filtering recommendations.

In this chapter we consider information about the user's personality as arguably domain-independent, and thus a potential source of information for establishing relationships between user preferences on items from different domains. Upon this assumption, we propose matrix factorization models for cross-domain collaborative filtering. In Section 5.1 we motivate and position our approach to personality-based recommendation. In Section 5.2 we review previous work on relating user preferences with personality traits, and exploiting the users' personality for recommendation purposes. Next, in Section 5.3 we present our personality-based matrix factorization models, both for single- and cross-domain scenarios, and in Section 5.4 we describe the experiments conducted to evaluate the models. Finally, in Section 5.6 we provide some conclusions from the results achieved.

5.1 INTRODUCTION

Previous research has shown that, in certain domains, people with similar personality traits are likely to have similar preferences (Cantador et al., 2013; Chausson, 2010; Rawlings and Ciancarelli, 1997; Rentfrow et al., 2003), and that correlations between user preferences and personality traits allow enhancing user profiles and improving personalized recommendations (Hu and Pu, 2011; Tkalcic et al., 2011). Motivated by these observations, in this thesis we aim to investigate the exploitation of user personality to address the cold start problem, compensating the lack of preferences of completely new users.

Proposed personality-based matrix factorization model

The majority of proposed personality-based recommendation approaches has mainly focused on the rating prediction task. Here, in contrast, we aim to address the item ranking task, which is appropriate when user preferences are not in the form of numeric ratings, such as the *thumbs up/down* in YouTube and the *likes* in Facebook. Moreover, existing approaches, e.g., those by Hu and Pu, (2011) and Tkalcic et al., (2011), have investigated the use of personality in Nearest Neighbors heuristics for collaborative filtering. We propose instead a model-based approach that builds on the matrix factorization method by Hu et al., (2008). Specifically, we propose to incorporate additional latent feature vectors for personality factors, and perform a new training procedure based on the *Alternating Least Squares* technique.

Non-overlapping users between domains

Rather than exploiting personality to directly estimate the unknown preferences, in the cross-domain recommendation setting we rely on personality factors to bridge the source and target domains, distinguishing two cases. In the first case, we consider the situation where there are no common users between the domains, and the transfer of knowledge is performed through shared personality factors. This approach is based on the hypothesis that certain combinations of personality factors correlate with the probability of the users' preferences over the items. For instance, users "open to new experiences" are more likely to interact with more items.

Overlapping users between domains

In the second case, we assume that there are some users belonging to both the source and target domains, and extend our personality-based matrix factorization model to exploit their preferences in the source domain. Specifically, we include another set of latent variables that enhance the users' models on the target domain with their preferences on the source domain. The transfer of knowledge is thus supported by both the personality factors and the source domain item factors. Our goal in this scenario is to understand whether personality information is still beneficial for the cold start, or if only source domain preferences are worth to be exploited.

Achieved results

Conducting experiments on a large dataset in various domains, namely movies, music and books recommendations, we empirically show that the proposed personality-based models allow collaborative filtering to better tackle the cold start problem. This is especially true for completely new users with no preference profile at all, who are usually provided with non-personalized suggestions based on the popularity of the items. We show that these users can significantly benefit from the application of our approaches, which are able to generate personalized recommendations that boost precision in ranges from 6% to 94%, depending on the domain. We also show, however, that this benefit vanishes once a sufficient number of target domain preferences for the user becomes available. Finally, we provide further insight into our models by comparing them against state of the art *active learning* approaches that elicit preferences directly from the user before generating any recommendation.

5.2 BACKGROUND AND RELATED WORK

The recommendation models we propose in this chapter are based on the assumption that information about the users' personality can be leveraged to estimate their preferences for recommendation. In the next subsections we first review the *Five Factor Model*, one of the most popular models for representing personality. We then describe studies on the relationships between user preferences and personality factors, and finally discuss state of the art approaches that exploit information about the users' personality in heuristic-based methods for collaborative filtering.

5.2.1 *Personality factors and their relationships with user preferences*

The Five Factor model of personalty

In psychology literature, personality is described as a "consistent behavior pattern and interpersonal processes originating within the individual" (Burger, 2010), accounting for individual differences in people's emotional, interpersonal, experiential, attitudinal and motivational styles (John and Srivastava, 1999). Personality is a predictable and quite stable aspect that forms human behaviors, and several models have been proposed to characterize and represent human personality. Among them, the Five Factor model (FFM) (Costa and McCrae, 1992) is considered one of the most comprehensive, and has been mostly used to build personality-based user profiles (Hu and Pu, 2011).

The FFM introduces five broad dimensions —called factors or traits, and commonly known as the *Big Five*— to describe an individual's personality, namely *openness, conscientiousness, extraversion, agreeableness* and *neuroticism*, which are defined as follows:

- **Openness (OPE)**: from *cautious/consistent* to *curious/inventive*. It reflects a person's tendency to intellectual curiosity, creativity and preference for novelty and variety of experiences. A high score of openness entails strong degrees of imagination, artistic interest, emotionality, adventurousness, intellect and liberalism.
- **Conscientiousness (COS)**: from *careless/easy-going* to *organized/-efficient*. This factor reflects a person's tendency to show self-discipline and aim for personal achievements, and to have an organized (not spontaneous) and dependable behavior. A high score of conscientiousness entails strong degrees of self-efficacy, orderliness, dutifulness, achievement-striving and cautiousness.
- **Extraversion (EXT)**: from *solitary/reserved* to *outgoing/energetic*. This factor reflects a person's tendency to seek stimulation in the company of others – showing sociability, talkativeness and assertiveness traits –, and to put energy in finding positive emotions, such as happiness, satisfaction and excitation. A high

score of extraversion entails strong degrees of friendliness, gregariousness, activity level, excitement-seeking and cheerfulness.

- **Agreeableness (AGR)**: from *cold/unkind* to *friendly/compassionate*. This factor reflects a person's tendency to be kind, concerned, truthful and cooperative towards others. A high score of agreeableness entails strong degrees of morality, altruism, sympathy, modesty, trust, cooperation and conciliation.

- **Neuroticism (NEU)**: from *secure/calm* to *unconfident/nervous*. This factor reflects a person's tendency to experience unpleasant emotions, such as anger, anxiety, depression and vulnerability, and refers to the degree of emotional stability and impulse control. A high score of neuroticism entails strong degrees of hostility, social anxiety, depression, immoderation, vulnerability and impulsivity.

The measurement of the five factors is usually performed by assessing *items* that are self-descriptive sentences or adjectives, and are commonly presented to the subjects in the form of short questions. In this context, the International Personality Item Pool[1] (IPIP) is a publicly available collection of items for use in psychometric tests, and the 20-100 item IPIP proxy for Costa and McCrae's commercial NEO PI-R test (IPIP-NEO, see (Goldberg et al., 2006)) is one of the most popular and widely accepted questionnaires to measure the Big Five in adult men and women without overt psychopathology. Alternatively, approaches exist that attempt to infer the people's personality factors implicitly, e.g., by mining user generated contents in social media (Farnadi et al., 2016), and analyzing social network structure (Lepri et al., 2016).

Relationships between user preferences and personality

Personality influences how people make decisions (Nunes and Hu, 2012), and people with similar personality traits are likely to have similar tastes. For example, Rentfrow et al., (2003) investigated how music preferences are related with personality in terms of the FFM. They showed that *reflective* people with high openness usually have preferences for jazz, blues and classical music, and *energetic* people with high degree of extraversion and agreeableness usually appreciate rap, hip-hop, funk and electronic music. Rawlings and Ciancarelli, (1997) observed that openness and extraversion are the personality factors that best explain the variance in personal music preferences. They showed that people with high openness tend to like diverse music styles, and people with high extraversion are likely to have preferences for popular music. In the movie domain, Chausson, (2010) presented a study showing that people open to experiences are likely to prefer comedy and fantasy movies, conscientious individuals are more inclined to enjoy action movies, and neurotic people tend to like

1 International Personality Item Pool, http://ipip.ori.org

romantic movies. Odic et al., (2013) explored the relations between personality factors and induced emotions while watching movies in different social contexts (e.g., alone vs. with someone else), and observed different patterns in experienced emotions as functions of the extraversion, agreeableness and neuroticism factors. More recently, Braunhofer et al., (2015c) showed that exploiting personality information in collaborative filtering is more effective than exploiting demographic information, which is a more typical approach for dealing with the new user problem in recommender systems. In particular, they showed that exploiting even a single personality factor (out of the five factor) may lead to a considerable improvement in recommendation accuracy.

Extending the spectrum of analyzed domains, Rentfrow et al., (2011) studied the relations between personality factors and user preferences in several entertainment domains, namely movies, TV shows, books, magazines and music. They focused their study on five content categories: aesthetic, cerebral, communal, dark and thrilling. The authors observed positive and negative relations between such categories and some of the personality factors, e.g., they showed that aesthetic content relate positively with agreeableness and negatively with neuroticism, and that cerebral content correlate with extraversion. Cantador et al., (2013) also considered several domains (movies, TV shows, books and music), and presented a preliminary study on the relations between personality types and entertainment preferences. Analyzing a large dataset of personality factor and genre preference user profiles, the authors extracted personality-based user stereotypes for each genre, and inferred association rules and similarities between types of personality of people with preferences for particular genres. Finally, in the multi-domain scenario of the Web, Kosinski et al., (2012) presented a study revealing meaningful psychologically relations between user preferences and personality for certain websites and website categories.

More fine-grained representation of personality

We notice that, as mentioned in (Cantador et al., 2013), additional user characteristics, such as the user's age and gender, and more fine-grained personality representations such as those based on personality facets, e.g., the *imagination*, *artistic interests*, and *emotionality* facets for the *openness* factor, may be of importance when discovering relationships between user preferences and personality. In the reviewed paper and in this chapter, such type of characteristics are not taken into consideration, and are left for future investigation. We develop and evaluate our recommendation models upon the fact that there exist certain relationships between user preferences and personality, which can benefit collaborative filtering in the cold start as done in previous work, described in the next subsection.

5.2.2 *Personality-based collaborative filtering*

Personality in heuristic collaborative filtering methods

The existence of certain relationships between personality characteristics and user preferences has motivated earlier studies supporting the hypothesis that exploiting personality information in collaborative filtering is beneficial. Tkalcic et al., (2011) applied and evaluated user similarity metrics for heuristic-based CF: a typical rating-based similarity based on Euclidean distance with personality data (five factors), and a similarity based on a weighted Euclidean distance with personality data. Their results show that approaches using personality data may perform statistically equivalent or better than approaches based on only ratings, especially in cold start situations. In her PhD dissertation, Nunes, (2009) explored the use of a personality user profile composed of IPIP-NEO items and facets in addition to the Big Five factors, showing that fine-grained personality user profiles can achieve better recommendations. Following the findings of (Rentfrow et al., 2003), Hu and Pu, (2009, 2011) presented a CF approach that leverages the correlations between personality types and music preferences: the similarity between two users is estimated by means of the Pearson's correlation coefficient on the users' five factors scores. Combining this approach with a rating-based CF technique, the authors showed significant improvements over the baseline of considering only ratings data. Later, Roshchina, (2012) presented an approach that extracts five factors profiles by analyzing hotel reviews written by users, and incorporates these profiles into a nearest neighbor algorithm to enhance personalized recommendations.

In (Fernández-Tobías and Cantador, 2015) we compared the performance of user-based nearest neighbor recommendations exploiting personality against cross-domain preferences. Our findings on a limited dataset of item genres from multiple domains (books, movies, music) showed that cross-domain information is in general preferable, although personality is still beneficial in several situations. We also observed that personality can be used together with auxiliary source preferences to further enhance cross-domain recommendations, serving as a motivation for the models proposed in this chapter. Moreover, the evaluation in that paper did not focus on cold start scenarios, and rather on the overall performance of the system, as opposed to the results presented in this chapter.

Personality in matrix factorization collaborative filtering models

It is worth noting that the above mentioned works on personality-aware CF make use of heuristic-based methods to compute user similarities and item rating estimations. Differently from them, in this chapter we propose a matrix factorization CF model –which has been shown to be in general more effective than heuristic approaches– that integrates the users' rating data and personality information. Moreover, with respect to previous work, the experimental study presented here has been conducted on much larger datasets composed

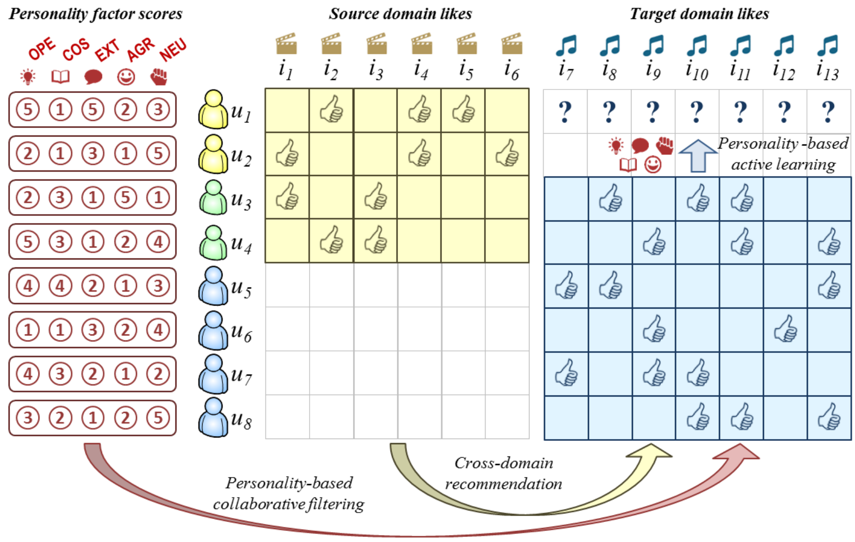

Figure 5.1: Exploiting user personality and cross-domain preferences to address the cold start problem. Note that the cold start user u_1 has not yet provided preferences in the target domain.

of positive-only feedback in the form of *likes*, rather than ratings, on several domains. Specifically, as described in Section 5.4, our dataset consists of 159,551 users and 16,303 items in the movies, music and books domains, while in (Hu and Pu, 2009, 2011) the considered data set contains only 111 users, in (Nunes, 2009) and (Roshchina, 2012) it is around 100, and in (Tkalcic et al., 2011) 52, all of which contain only a very limited number of items. Finally, in the chapter we shall show a more diverse set of results, observing that the users' personality is not equally useful in all the considered domains. For instance, the usage of personality in the movies and music domains will yield higher precision compared to the books domain.

5.3 PROPOSED RECOMMENDATION MODELS

The recommendation models we propose in this chapter are based on the hypothesis that information about the users' personality is available and can be used to address the cold start in collaborative filtering. First, we propose a novel extension of the matrix factorization method for positive-only feedback (e.g., click-through data, browsing history, item consuming counts) that is capable of exploiting auxiliary personality information for recommendation. Then, we present an adaptation of our model to the cross-domain scenario, so that personality factors allow transferring knowledge across domains when there are no common users, and later we further extend the model to incorporate source domain preferences when there is user overlap between domains.

Considered cross-domain scenario

Figure 5.1 illustrates the recommendation scenario we consider in this chapter, where a user u_1 has just registered into the system seeking for recommendations in a target domain, i.e., music in our example. We assume that personality information from all users is available in the form of Five Factor Model scores, and that maybe some users in the target domain also have preferences in a related source domain available, e.g., movies. In order to provide music recommendations for u_1, our matrix factorization model exploits u_1's personality information and movie preferences together with those from other users in a collaborative filtering fashion. The figure also shows the active learning approach, where personality and target domain preferences are used to select a subset of items that the user will be asked to evaluate before providing her with any recommendations. In our experiments we shall provide a comparison between our cross-domain model, and state of the art active learning approaches as a solution to the cold start problem.

5.3.1 *Personality-based matrix factorization for positive-only user feedback*

We now describe the proposed matrix factorization model extended with personality factors. First, let $\mathcal{U}, \mathcal{I}$ be the sets of users and items registered in the system, respectively, and let $\vec{p}_u \in \mathbb{R}^k, \vec{q}_i \in \mathbb{R}^k$ be latent feature vectors for user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$. As explained in Section 2.3.2, the user u's preference score towards item i in the standard matrix factorization is estimated as follows:

$$s(u, i) = \langle \vec{p}_u, \vec{q}_i \rangle \tag{5.1}$$

A list of recommended items for user u is generated by sorting the items in $\mathcal{I}$ by decreasing order of estimated preference, eventually ignoring those that the user has already rated.

Matrix factorization-based recommendation has been extensively studied in the literature, and it is known to yield inaccurate item relevance predictions in cold start situations. When little information about the user is known, the learned parameter $\vec{p}_u$ is unlikely to properly model the user's latent preferences, and for users completely new to the system this method is simply unable to compute any rating prediction. In our adaptation, we overcome this limitations by introducing additional parameters to model the user's personality.

Discretization of personality factor scores

Among the existing approaches for representing personality, in this thesis we focus on the Five Factor Model. As explained in Section 5.2.1, in the FFM the personality of each user is described using five independent dimensions or factors, namely *openness*, *conscientiousness*, *extraversion*, *agreeableness* and *neuroticism*. A user's personality profile is thus represented with a score for each factor, typically a real number in the range such as $[1, 5]$. In order to use this information we follow the same strategy as in (Elahi et al., 2013), mapping the five factors to a fixed set of Boolean attributes $\mathcal{A}$. Specifically, let $\vec{u} = (ope_u, con_u, ext_u, agr_u, neu_u)$ be the vector representation of user u's FF scores. We first discretize each score, and then map it to a different attribute depending on its value and factor for computing the set of user-specific attributes $\mathcal{A}(u)$. In Section 5.5.1 we will describe several discretization schemes we tested to compute $\mathcal{A}(u)$.

Personality-based user latent features

Once the user's personality factor scores are transformed, we modify Equation 5.1 to take personality information into account when computing item relevance predictions. Specifically, we define new additional latent feature vectors $\vec{y}_a \in \mathbb{R}^k$ for each attribute $a \in \mathcal{A}$. Now, the users are not only modeled in terms of their preferences, but also considering their personality attributes:

$$s(u,i) = \left\langle \vec{p}_u + \sum_{a \in \mathcal{A}(u)} \vec{y}_a, \vec{q}_i \right\rangle \quad (5.2)$$

An important characteristic of this model is that it is capable of generating item relevance values even if the user is completely new to the system, making it ideal for cold start situations. In such cases, the vector $\vec{p}_u$ is ignored and user preferences are estimated only on the basis the above attributes.

Personality-based matrix factorization

The prediction model defined in Equation 5.2 is inspired by the well known and widely used SVD++ model (Koren, 2008). SVD++ incorporates implicit feedback by introducing latent feature vectors for items rated by the user, whereas in Equation 5.2 the user's profile is augmented with latent feature vectors that model the user's personality. Unlike (Elahi et al., 2013) and (Koren, 2008), the method we propose here is intended for the top N recommendation task in the presence of positive-only user feedback rather than for the rating prediction task. We argue that positive-only feedback is more common in real applications, where users are usually not inclined to explicitly evaluate the items. Click-through data, browsing history, or item consuming counts are instead more easily acquired by the system, without requiring any effort of the user. However, it must be taken into account that in this setting, information about the users' *dislikes* is not available, and the fact that a user did not select a particular item might either indicate that the item is unknown to her or that she actually dislikes it.

Alternating Least Squares for personality-based matrix factorization

In order to deal with this type of user feedback, we follow the approach of Hu et al., (2008), where the matrix factorization method was adapted for positive-only feedback. In their model, predictions are still computed using Equation 5.1, but unlike standard MF that learns the model parameters by only exploiting the observed ratings, Hu et al.'s method also considers the not observed ones. Moreover, they argue that in this case the commonly used Stochastic Gradient Descent algorithm is no longer efficient, and propose an alternative optimization procedure based on *Alternating Least Squares* (ALS). In our personality-based model we incorporate the same learning technique, but for a different prediction model, namely that presented in Equation 5.2. Finally, the resulting loss function penalizes prediction errors over all possible user-item pairs, not only those for which an interaction was observed, and includes the additional model parameters for the personality factors:

$$\mathcal{L}(\mathbf{P},\mathbf{Q},\mathbf{Y}) = \sum_u \sum_i c_{ui}\left(x_{ui} - s(u,i)\right)^2 + \lambda\left(\|\mathbf{P}\|^2 + \|\mathbf{Q}\|^2 + \|\mathbf{Y}\|^2\right) \tag{5.3}$$

where $x_{ui} = 1$ if user u consumed (i.e., clicked, liked, purchased) item i, and $x_{ui} = 0$ otherwise. $s(u,i)$ is the item relevance prediction computed using Equation 5.2. Each row of the matrices $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}|\times k}, \mathbf{Q} \in \mathbb{R}^{|\mathcal{I}|\times k}, \mathbf{Y} \in \mathbb{R}^{|\mathcal{A}|\times k}$ contains the latent feature vector of a user, an item and an attribute, respectively. The confidence parameter c_{ui} controls how much the model penalizes mistakes in the prediction of x_{ui}, and is set to $c_{ui} = 1 + \alpha k_{ui}$ as proposed in (Hu et al., 2008). k_{ui} represents user u's feedback for item i, which is binary in the case of clicks and *likes*, or a positive number e.g., for item consuming counts, and is set to $k_{ui} = 0$ in the case that no interaction was observed. The constant α models the increase in confidence for observed feedback. Finally, the regularization parameter $\lambda \in \mathbb{R}^+$ is used to prevent overfitting.

The model parameters $\mathbf{P},\mathbf{Q}$ and $\mathbf{Y}$ are automatically learned by minimizing the loss function over all the user-item training pairs. We extend the method of Hu et al., (2008), deriving an ALS-based algorithm with an extra step for the additional $\mathbf{Y}$ parameters of the minimization problem defined in Equation 5.3. ALS is based on the observation that when all the parameters but one are fixed, Equation 5.3 becomes a standard least-squares problem with a solution that can be explicitly computed. First, we fix $\mathbf{Q}$ and $\mathbf{Y}$, and solve the optimization problem analytically for each $\vec{p}_u$ by setting the gradient to zero:

$$\vec{p}_u = \left(\mathbf{Q}^\mathsf{T}\mathbf{C}^u\mathbf{Q} + \lambda\mathbf{I}\right)^{-1}\mathbf{Q}^\mathsf{T}\mathbf{C}^u\left(\vec{x}(u) - \mathbf{Q}\textstyle\sum_{a\in\mathcal{A}(u)}\vec{y}_a\right) \tag{5.4}$$

where $\mathbf{C}^u$ is a $|\mathcal{I}| \times |\mathcal{I}|$ diagonal matrix such that $\mathbf{C}^u_{ii} = c_{ui}$, $\vec{x}(u)$ is a column vector with all the x_{ui} values for user u. Let for simplicity $\vec{z}_u = \vec{p}_u + \sum_{a\in\mathcal{A}(u)}\vec{y}_a$. We then fix $\mathbf{P}$ and $\mathbf{Y}$ and optimize each $\vec{q}_i$ in a similar fashion:

$$\vec{q}_i = \left(\mathbf{Z}^\mathsf{T}\mathbf{C}^i\mathbf{Z} + \lambda\mathbf{I}\right)^{-1}\mathbf{Z}^\mathsf{T}\mathbf{C}^i\vec{x}(i) \tag{5.5}$$

Analogously, $\mathbf{C}^i$ is a $|\mathcal{U}| \times |\mathcal{U}|$ diagonal matrix where $\mathbf{C}^i_{uu} = c_{ui}$, $\vec{x}(i)$ is a column vector with all the x_{ui} values, and the matrix $\mathbf{Z}$ contains the $\vec{z}_u$ vectors as rows. Finally, we fix $\mathbf{P}$ and $\mathbf{Q}$, and optimize for each $\vec{y}_a$:

$$\vec{y}_a = \left[\mathbf{Q}^\mathsf{T}\left(\sum_{u\in\mathcal{U}(a)}\mathbf{C}^u\right)\mathbf{Q} + \lambda\mathbf{I}\right]^{-1}\sum_{u\in\mathcal{U}(a)}\mathbf{Q}^\mathsf{T}\mathbf{C}^u\left(\vec{x}(u) - \mathbf{Q}\vec{z}_{u\setminus a}\right) \tag{5.6}$$

Algorithm 3 ALS for personality-based matrix factorization

procedure TRAIN
  Initialize $\mathbf{P}, \mathbf{Q}, \mathbf{Y}$ at random
  for $iteration \leftarrow 1, ..., T$ **do**
    P step
      Fix $\mathbf{Q}, \mathbf{Y}$ and optimize all $\vec{p}_u$ in parallel using Eq. (5.4)
    Q step
      Fix $\mathbf{P}, \mathbf{Y}$ and optimize all $\vec{q}_i$ in parallel using Eq. (5.5)
    Y step
    ▷ Computation of attribute vectors cannot be parallelized
    for all $a \in A$ **do**
      Fix $\mathbf{P}, \mathbf{Q}, \vec{y}_{b \neq a}$ and optimize $\vec{y}_a$ using Eq. (5.6)
    end for
  end for
end procedure

where $\mathcal{U}(a) = \{u \in \mathcal{U} \mid a \in \mathcal{A}(u)\}$ is the set of users that have attribute a, and $\vec{z}_{u \setminus a} = \vec{p}_u + \sum_{b \in \mathcal{A}(u), b \neq a} \vec{y}_b$ is defined as before but without including attribute a. Note that, unlike the case of user and item attributes, re-computing an attribute vector depends on the current state of all the other attribute features through the $\vec{z}_{u \setminus a}$ vector.

In the training process, we alternate between three steps fixing a different set of latent feature parameters each time. This process is repeated for a fixed number of iterations T, as depicted in Algorithm 3. Finally, once the training stage is completed, we use the learned parameters to compute item relevance predictions for the test users using Equation 5.2. For each user, we estimate all the scores for unknown items and sort them in descending order. The top ten items of the list are recommended to the user as the more likely to be relevant.

Computational complexity of personality-based matrix factorization

Similarly to the model of Hu et al., (2008), the complexity of our personality-aware MF method is $\mathcal{O}(k^3|\mathcal{U}| + k^2|\mathcal{R}_+|)$ for the P-step and $\mathcal{O}(k^3|\mathcal{I}| + k^2|\mathcal{R}_+|)$ for the Q-step, where $|\mathcal{R}_+|$ is total number of observed user-item preferences. Here we have used an optimization described in (Hu et al., 2008) to reduce the complexity from $|\mathcal{U}| \cdot |\mathcal{I}|$ to $|\mathcal{R}_+|$ terms. We refer the reader to that paper for more details. In these steps, the latent feature vectors can be easily computed in parallel within each step. The main computational cost relies on the Y-step, in which we have to iterate over the whole $\mathcal{U}(a)$ set for each attribute. Updating all the attribute vectors has complexity $\mathcal{O}(k^3|\mathcal{A}| + k^2|\mathcal{A}|(|\mathcal{I}| + |\mathcal{R}_+|))$, with the drawback that it cannot be parallelized since the recomputation of each attribute vector depends on the current state of the others. We note, however, that the number of attributes $|\mathcal{A}|$ is usually small, and the overhead required by the additional latent features is acceptable, making the complexity of our algorithm comparable to that of standard ALS-based MF. Also, as we shall see in Section 5.5.1, we consider at most five attributes for each user, one for each dimension of the FFM, so $|\mathcal{A}(u)| \leqslant 5$, and recommendations are fast to compute.

5.3.2 *Personality-based cross-domain collaborative filtering*

In the previous section we have presented our personality-based extension of matrix factorization, which can be applied to address the cold start problem in single-domain recommendation scenarios. We now turn our attention to the cross-domain setting. First, we describe how the previous model can be applied to the case when there are no common users between domains, and next we further extend the model to deal with additional user feedback when there is user overlap.

Scenario I: No user overlap between domains

The application of our personality-based matrix factorization model is based on the assumption that, even when there are neither users nor items common to source and target domains, personality information can be exploited to bridge user preferences across the domains in a similar fashion as the social tag-based models presented in Chapter 4.

In particular, we consider the concatenation of the user-item matrices from the source and target domains $\mathcal{R} = \mathcal{R}_S \cup \mathcal{R}_T$, and directly train our personality-based model by minimizing Equation 5.3 to jointly factorize both domains, letting now $u \in \mathcal{U}_S \cup \mathcal{U}_T$ and $i \in \mathcal{I}_S \cup \mathcal{I}_T$. Again, as for the TagGSVD++ model presented in Chapter 4, a given (u, i) pair can be univocally identified in either $\mathcal{R}_S$ or $\mathcal{R}_T$ since there is no user or item overlap.

Personality factors for transferring knowledge between domains

The underlying assumption for this approach is that the presence or absence of certain personality factors may correlate with the likelihood that a user interacts with an item, e.g., users with a high *openness* factor are more likely to interact with more items, and that this correlation can be transferred across domains. Hence, personality factors allow bridging the domains through the attributes shared between $\mathcal{A}_S$ and $\mathcal{A}_T$. Moreover, personality information is inherent to the users, and arguably domain-independent: if the users of both domains $\mathcal{U}_S$ and $\mathcal{U}_T$ are sampled from the same population, we can further assume that their Five Factor scores will follow similar distributions in each domain, so that the same personality attributes are found in both domains, i.e., $\mathcal{A}_S = \mathcal{A}_T \triangleq \mathcal{A}$. In this setting, the transfer of knowledge between the source $\mathcal{S}$ and the target $\mathcal{T}$ is supported by the personality latent factors $\vec{y}_a, a \in \mathcal{A}$, which are the only parameters involved in both domains.

Scenario II: User overlap between domains

Personality factors for enhancing source domain user preferences

In the case that some users are present in both domains, the cross-domain recommendation model aims to also exploit the user's source domain preferences in order to provide the user with better item suggestions in the target domain. We hypothesize that personality information can be leveraged to enhance cross-domain recommendations by enriching user profiles not only with preferences from auxiliary domains but also with Big Five scores.

In order to understand the contribution of personality factors in the cross-domain setting with user overlap, we first adapt the personality-based matrix factorization model proposed in Section 5.3.1 by replacing personality attributes with cross-domain user preferences. Let $\mathcal{S}$ be the source domain, $\mathcal{T}$ the target domain, and $\mathcal{I}_S, \mathcal{I}_T$ their respective sets of items. We estimate the user u's preference for item $i \in \mathcal{I}_T$ as

$$s(u,i) = \left\langle \vec{p}_u + \sum_{j \in \mathcal{I}_S(u)} \vec{y}_j, \vec{q}_i \right\rangle \tag{5.7}$$

where $\mathcal{I}_S(u)$ is the set of items in the source domain for which user u expressed a preference. This method is a simple extension of SVD++ (Koren, 2008) that expands the user's latent representation in the target domain $\vec{p}_u$ with latent feature vectors modelling the effect of user feedback in a source domain. Another difference relies on the training algorithm, which is here based on ALS instead of stochastic gradient descent, as described in Algorithm 3. It is worth noting that in order for this model to be successful, the sets of users from the source and target domains must overlap, i.e., $\mathcal{U}_S \cap \mathcal{U}_T \neq \emptyset$. Even when there are users with preference data in both domains, the preferences from the source domain may not be relevant for recommendation in the target domain, which is another limitation of the approach. Intuitively, user *likes* from a source domain such as restaurants may not be indicative of the user's tastes on an unrelated domain such as music.

We then combine both user personality and source domain preferences into a common set of user attributes, aiming to understand if personality information can be used to enhance cross-domain recommendations in the cold start, or if, on the other hand, only cross-domain preferences are useful. Specifically, we predict item relevance values as follows:

$$s(u,i) = \left\langle \vec{p}_u + \sum_{a \in \mathcal{A}(u)} \vec{y}_a + \sum_{j \in \mathcal{I}_S(u)} \vec{y}_j, \vec{q}_i \right\rangle \tag{5.8}$$

The above model is also trained using the ALS technique described in Algorithm 3, and despite its simplicity we believe it is, to the best of our knowledge, the first attempt to enhance cross-domain matrix factorization with personality information. We note that, differently from the personality-based model for single-domain recommendation from Section 5.3.1, the number of parameters here is much larger, which has a direct impact on the complexity of the learning process. We are nevertheless interested in comparing the benefits of personality information and cross-domain preferences for new users, and thus utilize the same recommendation model for both.

5.4 EXPERIMENTS

In this section we explain the experimental work conducted to evaluate our personality-based recommendation models. We first describe the dataset we used, and the processing we performed on it. We then explain the followed evaluation methodology for the cold start setting, and the considered baseline recommendation approaches. Finally, we report and analyze the achieved results in single- and cross-domain recommendation scenarios, and conclude with a discussion comparing the evaluated solutions to the cold start problem.

5.4.1 *Dataset*

The myPersonality project

The dataset used in our experiments is part of the database made publicly available in the myPersonality project[2] (Bachrach et al., 2012). myPersonality is a Facebook application in which users take psychometric tests, and receive feedback on their personality factor scores. The users allow the application to record personal information from their Facebook profiles, such as demographic and geo-location data, *likes*, status updates, and friendship relations, among others. In particular, as of October 2016, the tool instantiated a database with 36 million Facebook *likes* of 3 million users for items of diverse nature – people (actors, musicians, politicians, sportsmen, writers, etc.), objects (movies, TV shows, songs, books, video games, etc.), organizations, events, etc.– and the Big Five scores of 7.5 million users, collected using 20 to 336 item IPIP questionnaires.

Data processing

Due to the size and complexity of the database, in this chapter we restrict our study to a subset of it. Specifically, we selected the *likes* assigned to the items belonging to one of the following three domains: books, movies and music. To determine which items in the original database belong to each of such domains, we used Facebook item categorization data. Specifically, we manually identified certain categories for each domain, e.g., *Music genre, Musician/Band, Album* and *Song* for the music domain. Such categories were not always assigned correctly. For instance, there were many music *Albums* annotated with the *Musician/Band* category. Moreover, the names of the items were not always correct, e.g., some of them contained misspellings, and often were not used in a single, concise way, e.g., they

2 The myPersonality project, `http://mypersonality.org`

were given in terms of morphological deviations, such as `science fiction`, `science-fiction`, `sci-fi` and `sf`.

In order to address the above issues –checking misspellings, unifying morphological deviations, and rectifying categorizations– we performed a number of transformations that consolidated incorrect and duplicate items with correct ones, while exploiting external knowledge to set the items categories. Since it is outside of the focus of this work, we do not enter into details about the mentioned data transformations. We just mention that such operations were proposed in previous work (Cantador et al., 2010a; Szomszor et al., 2008b), and have been validated by automatically mapping the processed names of the items with the URIs of entities in DBpedia[3] (Lehmann et al., 2015) (the Wikipedia ontology) via SPARQL[4] queries; we discarded those items that could not be mapped to DBpedia entities. For instance, in the music domain, those items whose names were consolidated as `mozart`, were mapped to `http://dbpedia.org/page/Wolfgang_Amadeus_Mozart`, and maintained as a single item in the final dataset. In Chapter 6, which is dedicated to the exploitation of item semantic metadata for cross-domain recommendation, we shall detail the method we implemented for mapping items to DBpedia entities.

Dataset statistics

The whole process was conducted on the 6,500 most popular items in the dataset, i.e., the items with highest numbers of *likes*. Note that this may favor the good performance of popularity-based recommendation methods, as we shall observe in following sections. The final dataset is described in Table 5.1. It consists of 5,027,593 *likes* from 159,551 users on 16,303 items. Its minimum, maximum and average (standard deviation) numbers of *likes* per user are 1, 164 and 3.87 (4.46) for books, 1, 741 and 13.02 (18.78) for movies and 1, 648 and 19.49 (28.80) for music. We note that in order to be able to evaluate the effectiveness of using personality on users with various degrees of coldness (i.e., containing different numbers of *likes*), only users that entered a minimum of 20 *likes* where considered. After that, there were 1,208 users in the book domain, out of which 1,200 (99.34%) and 1,190 (98.51%) had at least one preference in the movie and music domains, respectively; 26,951 users in the movie domain, out of which 23,826 (88.40%) and 26,810 (99.48%) had also preferences in the book and music domains, respectively; and finally, 43,702 users in the music domain, out of which 34,215 (78.29%) and 43,134 (98.70%) with also book and movie preferences, respectively.

---

3 The DBpedia knowledge repository, `http://http://dbpedia.org`
4 SPARQL Query Language for RDF, `http://www.w3.org/TR/rdf-sparql-query`

Table 5.1: Statistics of the used dataset

Domain	Books	Movies	Music
Users	91 854	141 123	145 476
Items	4543	5389	6371
Likes	355 112	1 837 152	2 835 329
Users with $\geqslant$ 20 likes	1208	26 951	43 702
Avg. likes/user	3.87	13.02	19.49

5.4.2 *Evaluated approaches*

Evaluated recommendation approaches

We compared the performance of our personality-based matrix factorization models against the following recommendation approaches:

- **Most popular**. Non-personalized method that recommends the most popular items that the user has not already liked. The popularity of an item is measured as the number of users in the training set who liked it.

- **iMF**. Matrix factorization method for positive-only feedback by Hu et al., (2008). We note that this method is unable to compute item relevance predictions for completely new users with no preference information. In our experiments, we set the number of factors $k = 10$, the regularization parameter $\lambda = 0.015$, and the confidence parameter $\alpha = 1$.

- **Active learning**. We adapted the personality-based preference elicitation method proposed by Elahi et al., (2013) to use our matrix factorization model. Rather than directly computing a list of recommendations, this method first predicts a set of candidate items that the user is likely to know, and then asks the user to provide feedback on them. The collected feedback is added to the user's profile to re-train the recommendation model with the new information. Active learning represents an alternative approach to the cold start problem, by acquiring user preferences, instead of exploiting auxiliary information as in our models for single and cross-domain recommendation. We therefore report its results only when we compare all the approaches proposed to the cold start against each other in Section 5.5.5.

- **Personality MF**. Our proposed extension of iMF, which exploits personality information. The mapping of Five Factor scores into attributes is described in Section 5.5.1. We chose the same parameters as for iMF for better comparison, although preliminary tests did not show significant difference using other values: $k = 10, \lambda = 0.015$, and $\alpha = 1$.

We did not evaluate other general-purpose recommendation approaches for positive-only feedback from the state of the art, as they have the same limitation than iMF and are not available to compute item relevance predictions for completely new users. Furthermore, as we shall show, iMF is a good enough model for the cold start once some user feedback is available.

Regarding cross-domain approaches, we observed that only the method proposed in (Hu et al., 2013) is suitable for positive-only feedback, but it is not designed to handle other auxiliary information such as personality. Since we are interested in analyzing the quality of cold start recommendations with and without exploiting personality, it is preferable to use the same recommendation model in both cases. Hence, we limited our study to the algorithm proposed in Section 5.5.4. Moreover, the method presented in (Hu et al., 2013) needs users common to the domains, and cannot deal with the no overlap scenario.

5.4.3 *Evaluation methodology and metrics*

The evaluation of the proposed models was conducted utilizing a modified user-based 5-fold cross-validation strategy, based on the methodology by Kluver and Konstan, (2014) for cold start evaluation that we reviewed in Section 2.4.2.

Data split

Our goal is to understand how the different approaches perform as the number of observed *likes* in the target domain increases. First, we divide the set of users into five subsets of roughly equal size. In each cross-validation stage, we keep all the data from four of the groups in the training set. Then, for each user u in the fifth group –the test users– we randomly split her *likes* into three subsets, as depicted in Figure 5.2:

1. A *training set*, initially empty and incrementally filled with u's *likes* one by one to simulate different cold start profile sizes,
2. A *validation/candidate set* containing the set of *likes* to be elicited by the active learning strategies or for tuning hyperparameters, and
3. A *testing set* used to compute the performance metrics.

The above procedure was modified for the cross-domain scenario by extending the training set with the full set of *likes* from the auxiliary domain, in order to obtain the actual training data for the predictive models. Similarly, this evaluation strategy was further modified to measure the performance of the active learning strategy. In particular, the evaluation of an active learning method for a specific user profile size closely follows the evaluation approach proposed by Elahi et al., (2014), and proceeds in the following way:

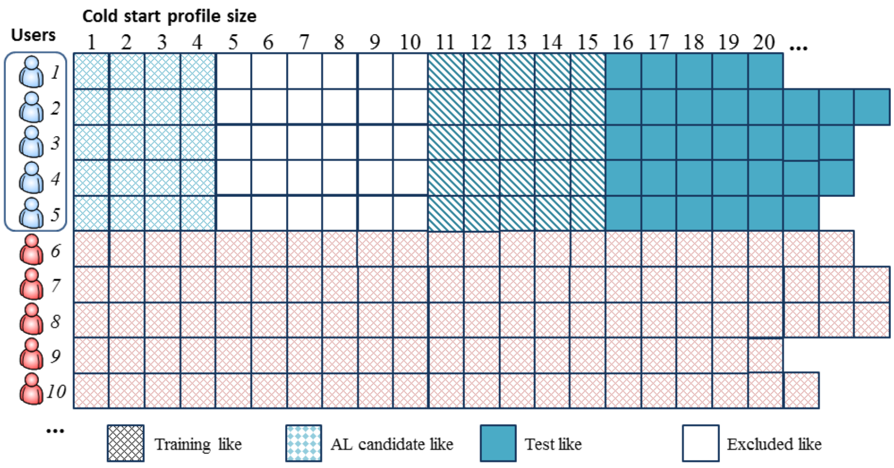

Figure 5.2: Overview of the cold start evaluation setting in a given cross-validation fold. The box indicates the test users in the current fold, whose profiles are split into training, candidate, and testing sets. Different cold start profle sizes are simulated by sequentially adding *likes* to their training sets —four in the figure.

1. The performance metrics are measured on the testing set, after training the prediction model on the training set.
2. For each user in the testing set:
   a) Using the active learning method, the top $N = 5$ candidate items that are not yet in the training set are selected for preference elicitation.
   b) Assign to the training set the user's likes for these selected items that are also found in the candidate set, if any.
3. The performance metrics are measured again on the testing set, after re-training the prediction model on the new, extended training set.

Precision metrics

We adopted three widely used accuracy and ranking metrics for collaborative filtering with positive-only feedback (Yao et al., 2014), namely *Mean Average Precision* (MAP), *Half-Life Utility* (HLU) and *Mean Percentage Ranking* (MPR).

- MAP measures the overall performance based on precision at different recall levels (see Equation 2.31). Larger values of MAP correspond to better recommendation performance.
- HLU measures the utility of a recommendation list for a user, with the assumption that the likelihood that the user will choose a recommended item decays exponentially with the item's ranking (Breese et al., 1998). A larger HLU means better performance.

- MPR estimates the user satisfaction of items in a ranked recommendation list, and is calculated as the mean of the percentile ranking of each test item within the ranked list of recommended items for each test user (Hu et al., 2008). It is expected that a randomly generated recommendation list has MPR close to 50%. A smaller MPR corresponds to a better recommendation performance.

In our experiments we observed an equivalent behavior of the approaches in terms of MAP, HLU, and MPR. Hence, for brevity, we only report MAP values in the analysis presented in Section 5.5.

Novelty and coverage metrics

We also computed two metrics for assessing aggregate item novelty and catalog coverage, namely the *AveragePopularity* and *Spread* metrics.

- *AveragePopularity* globally measures the mean popularity of the recommended items across the ranked lists of the users (Ziegler et al., 2005). It is expected that users prefer lists containing more novel (less popular) items. However, if the presented items are too novel, then the user is unlikely to have any knowledge of them, and will not be able to understand or *like* them. Hence, moderate values indicate a better performance (Kluver and Konstan, 2014).

- *Spread* measures the catalog coverage (or aggregate item diversity) of the recommended items, computed as the entropy of the distribution of items across the recommendation lists of all users. Maximal spread corresponds to uniform distributions where all items are suggested equally often, whereas low values mean that the algorithm is frequently recommending the same set of items. It is assumed that algorithms with a good understanding of the users are able to suggest different users with different items. However, it is not expected to achieve a perfect spread without making avoidably bad recommendations. Hence, moderate values are preferable (Kluver and Konstan, 2014).

For each cold start profile size, we built the recommendation models using the data in the final training set. Then, for each test user, we generated a ranked list of the top 10 suggested items from the set of items in the training set that are not yet known to the user. The performance is estimated from the output of each model and the test set using the above mentioned metrics. We note that in our evaluation, any item ranked after position 10 by the model is considered not relevant when computing the metrics, as we are interested in the more realistic setting where the user only examines a limited subset of the recommendations.

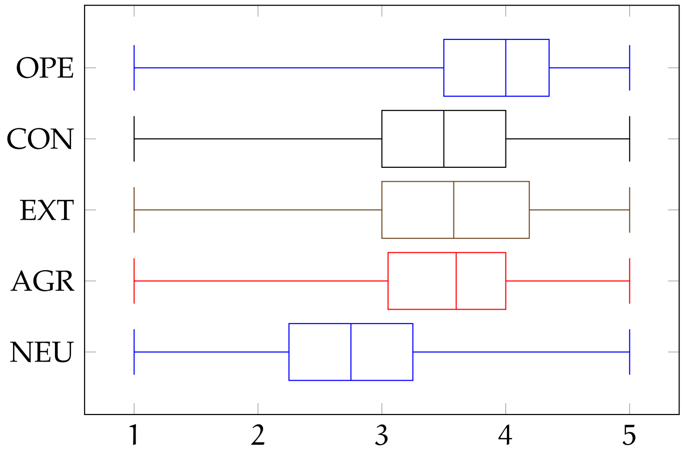

Figure 5.3: Distribution of Five Factor personality scores in our dataset.

5.5 RESULTS

Cold start scenarios

We now report and analyze the results obtained in our experiments. Throughout the section, we distinguish between two cold start scenarios:

- *Extreme* cold start, in which there are no *likes* at all from the active user, and recommendations are computed only on the basis of personality and/or cross-domain information.
- *Moderate* cold start, in which we assume that at least one *like* is given, and incrementally evaluate the performance of a recommendation model with larger and larger profile size of the active user. We thus aim to understand how the different approaches behave as the amount of available user preferences increases.

5.5.1 *Representing personality factors*

Tested methods for discretizing personality factor scores

In our first experiment we evaluated several discretization methods to define the user-specific personality attributes $\mathcal{A}(u)$ built from the Five Factor scores (see Section 5.3.1). In particular, we considered the following strategies to map the scores $\vec{u} = (\mathrm{ope}_u, \mathrm{con}_u, \mathrm{ext}_u, \mathrm{agr}_u, \mathrm{neu}_u)$ of user u from the $[1, 5]$ interval into Boolean attributes:

- Rounding (ROUND). The personality scores are rounded to the nearest integer, and an attribute is associated to each factor-value combination. For instance, a user with personality profile $\vec{u} = (2.3, 4.0, 3.6, 5.0, 1.2)$ will be assigned the set of attributes $\mathcal{A}(u) = \{\mathrm{ope}_2, \mathrm{con}_4, \mathrm{ext}_4, \mathrm{agr}_5, \mathrm{neu}_1\}$. Therefore, we consider 25 possible attributes in total, $|\mathcal{A}| = 25$, five for each personality factor: $\mathrm{ope}_1, \mathrm{ope}_2, \cdots, \mathrm{ope}_5, \mathrm{con}_1, \cdots, \mathrm{neu}_5$.

Table 5.2: Performance of the evaluated personality discretization methods.

Method	Books	Movies	Music
ROUND	**0.0790**	0.0630	0.0621
HALF	0.0738	0.0630	0.0616
2F	0.0765	0.0631	**0.0626**
2Q	0.0779	0.0637	**0.0626**
3F	0.0779	0.0627	0.0624
3Q	0.0779	**0.0639**	0.0622
MAX	0.0767	0.0636	0.0624
OPE	0.0765	0.0638	0.0607
CON	0.0761	0.0638	0.0613
EXT	0.0756	0.0630	0.0622
AGR	0.0774	**0.0639**	0.0620
NEU	0.0769	0.0626	0.0607

- Nearest half (HALF). We proceed exactly as before, but rounding to the nearest half-integer in steps of 0.5 instead of the nearest integer. This results in 45 possible attributes, $|\mathcal{A}| = 45$. In the previous example, $\mathcal{A}(u) = \{ope_{2.5}, con_4, ext_{3.5}, agr_5, neu_1\}$.

- Two levels, fixed (2F). The $[1,5]$ is divided for each factor in two levels, $L = [1,3]$ and $H = (3,5]$, resulting in $|\mathcal{A}| = 2 \times 5 = 10$ possible attributes. In our example, $\mathcal{A}(u) = \{ope_L, con_H, ext_H, agr_H, neu_L\}$.

- Two levels, quantiles (2Q). The 2F method assumes that the scores for all factors are distributed equally, which can result in suboptimal discretizations. In fact, Figure 5.3 shows that the scores have different distributions for each factor. Hence, the 2Q works similarly to 2F, except the intervals are split using the median value for each factor across the values in the dataset.

- Three levels, fixed (3F). Similar to 2F, but the intervals are split into 3 levels, $L = [1, 7/3], M = (7/3, 11/3]$, and $H = (11/3, 5]$, which results in $|\mathcal{A}| = 3 \times 5 = 15$ possible attributes. Following our example, $\mathcal{A}(u) = \{ope_L, con_H, ext_M, agr_H, neu_L\}$.

- Three levels, quantiles (3Q). Analogous to 3F, but the intervals are split based on tertiles for each factor.

- Max. Only the dominant factor is considered, independently of the value. This results in $|\mathcal{A}| = 5$ total attributes, one for each possible dominant factor. In our example, $\mathcal{A}(u) = \{agr\}$.

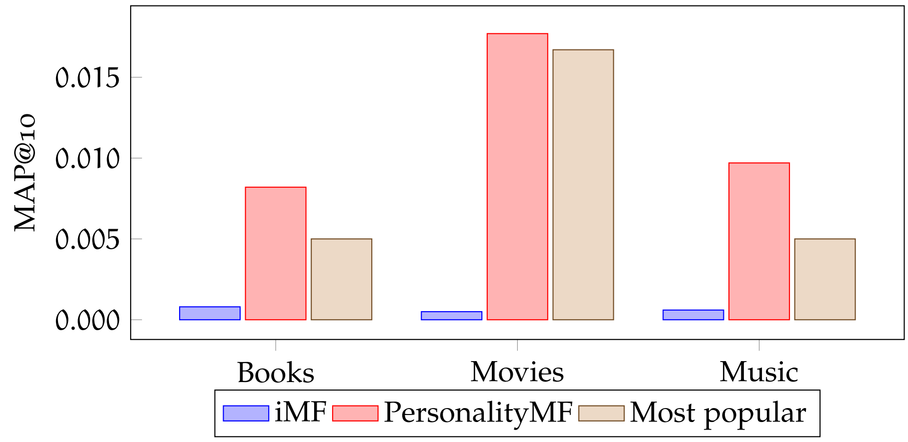

Figure 5.4: MAP@10 in the extreme cold start scenario.

- One factor (OPE, CON, EXT, NEU, AGR, NEU). Like ROUND, but only one of the Five Factors is considered each time, rounding the score to the nearest integer. This results in $|\mathcal{A}| = 5$ possible attributes for each instantiated factor.

Our goal was to empirically determine which of the previous discretization methods is more effective for our personality-based matrix factorization model. For such purpose, we trained the model separately in each domain and computed its performance in terms of MAP over the validation set, separately for each domain. In order to get a global estimate, we average the results across the different cold start levels. The obtained results are shown in Table 5.2, with the best values for each domain highlighted in bold.

We see that the ROUND method clearly achieves the best results in the books domain, while the performance in the movies and music domains is very similar for all the methods. Therefore, in the rest of our experiments we use ROUND to map the Five Factor scores into personality attributes. On a side note, we observe that no single factor (OPE, CON, EXT, AGR, NEU) is consistently better than the rest, indicating that none of them has more predictive power on its own.

5.5.2 *Exploiting personality for cold start single-domain recommendation*

Before addressing the cross-domain recommendation problem, the goal of the experiments reported in this section is to show if personality information can be used to improve the performance of matrix factorization in cold start situations on a single domain. Using the evaluation methodology described in Section 5.4.3, we computed HLU, MAP and MPR for different amounts of observed *likes* for items in the training set of the target domain.

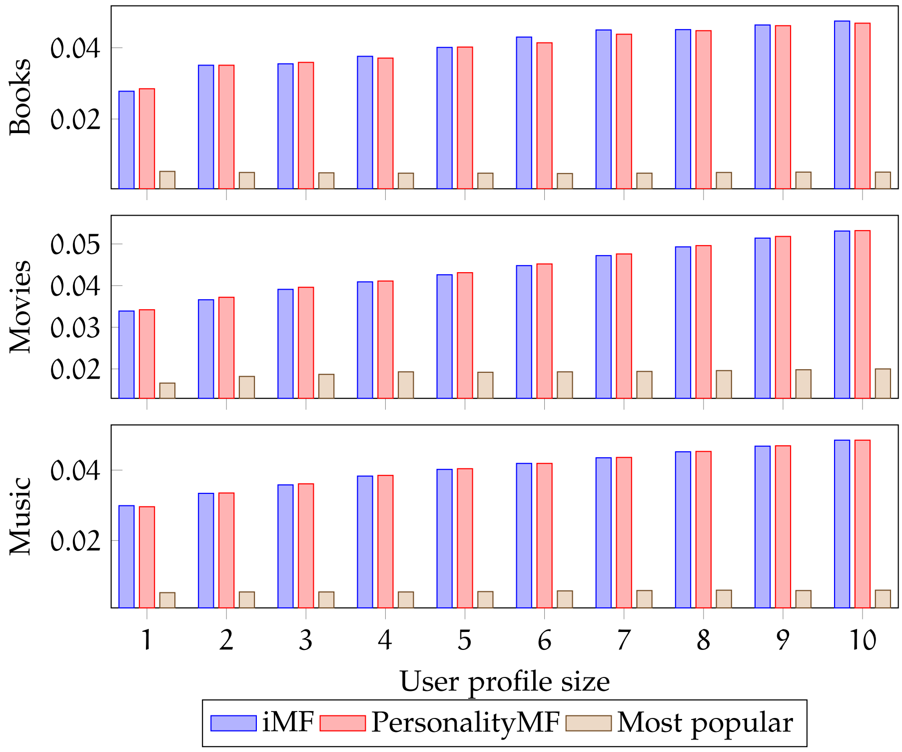

Figure 5.5: MAP@10 for different cold start user profile sizes.

Precision results in extreme cold start

We compare our proposed personality-based matrix factorization model (**Personality MF**), which computes item relevance predictions using Equation 5.2, against **iMF**, a state of the art method by Hu et al., (2008), which uses Equation 5.1 and does not exploit any auxiliary information. We also evaluate a non-personalized baseline, **Most popular**, that always recommends the most popular items. Results in terms of MAP@10 for the *extreme cold start* scenario are shown in Figure 5.4, for the three domains available in our dataset. The results for HLU and MPR were very similar, and therefore we do not report them here. We note that the small values obtained are due to the large item catalogs of our dataset. The set of possible candidate items to recommend for each test user is also large, leading to a low probability of matching a test item in the user's recommendation list.

In the figure we see that in all cases, Personality MF significantly outperforms iMF and the popularity baseline (Wilcoxon signed-rank test, $p < 0.05$). Our personality-based model is specially beneficial in the books and music domains, where it achieves relative improvements of 64% and 94%, respectively. The relationships between user preferences and personality seem to be stronger in these domains, although a more exhaustive analysis is required to confirm this observation. Nonetheless, we could conclude that personality information is highly beneficial in the extreme cold start situation, and that it is able to mitigate the total absence of user preferences, and recommend relevant items.

Precision results in moderate cold start

In Figure 5.5 we show the performance of the different approaches for increasing values of the user profile size in the cold start, again in terms of MAP@10. The Most popular baseline is clearly not a competitive approach, and the personalized approaches perform better as more ratings are available. We do not appreciate a significant difference in performance between iMF and Personality MF in any of the domains, indicating that personality information is not determinant once user preference data can be exploited.

Our results differ from those reported in (Hu and Pu, 2011), where it was shown that the user-based Nearest Neighbors method enhanced with personality clearly achieves better performance than using only ratings, for users with 2, 5, and 10 ratings in a music recommender system. It is worth noting that here we report results in a distinct, larger dataset (43,702 vs. 111 users, see Table 5.1) composed of *likes* (positive-only feedback) instead of numeric ratings. Also, we analyze the effects of integrating personality into the matrix factorization, instead of the nearest neighbor heuristic, and evaluate the performance for users completely new to the system.

We conclude that, in terms of accuracy, personality proves useful for completely new users in the three analyzed domains. In the other cases, iMF is competitive enough, and does not require any additional information. We argue, nonetheless, that the extreme cold start is a critical stage of a recommender system; the system must keep the user engaged, and exploiting personality is a good option to find relevant items for the user. Also, once some *likes* are observed, more subtle relations between user preferences and personality could be unveiled by taking into account additional variables by means of fine-grained representations of personality, as suggested in (Nunes, 2009).

Novelty and coverage results

In addition to accuracy, we also analyze the performance of the approaches in terms of item novelty and catalog coverage, as shown in Table 5.3. From the table, we observe similar behavior in all the considered domains: Personality MF and iMF on average recommend items with the same moderate popularity, except for completely new users. In that case, Personality MF recommends less novel items but still not simply the most popular ones –between 9.5% and 20% less popular on average, compared to the baseline. In terms of coverage, the personalized approaches recommend more varied items than the Most popular baseline, which always suggests the same set of items. We again see that without any available *likes*, personality-based MF approaches the behavior of the Most popular baseline. It is worth noting that in the extreme cold start situation the coverage of iMF is similar to Most popular, while Personality MF is much better in that respect.

Table 5.3: Novelty and coverage of collaborative filtering approaches in the cold start. Results for the *moderate* scenario with profile sizes 1–10 are stable, hence we report the average.

		Extreme cold start		Moderate cold start	
	Method	Avg. Pop.	Spread	Avg. Pop.	Spread
Books	iMF	7.12	3.32	142.80	6.29
	Personality MF	185.19	5.26	144.59	6.27
	Most popular	231.26	3.32	237.04	3.47
Movies	iMF	186.80	3.32	4056.75	6.43
	Personality MF	5717.94	4.75	4080.65	6.43
	Most popular	6447.28	3.32	6637.56	3.47
Music	iMF	311.36	3.32	6565.61	6.87
	Personality MF	9846.59	4.73	6592.58	6.86
	Most popular	10 877.38	3.32	11 113.77	3.45

5.5.3 *Exploiting personality for cross-domain recommendation without user overlap between domains*

In our next experiment we aim to analyze the effect of personality factors in cross-domain recommendtions when ther is no user and item overlap between the domains. As described in the first part of Section 5.3.2, the transfer of knowledge in this scenario is performed through the lantent factors corresponding to personality attributes. Therefore, our goal is to understand whether the effect of personality on the observed likes is worth being transferred across domains. For such purpose, we compare the following methods:

- Single-domain recommendation models from the previous section, namely iMF, PersonalityMF, and Most popular.
- Cross-domain personality-based matrix factorization, where the transfer of knowledge is performed through the personality factors. We refer to this approach as `Books`, `Movies` or `Music`, depending on the source domain that is exploited.

Precision results in the extreme cold start without user overlap

Figure 5.6 shows the obtained results for the *extreme* cold start scenario. In the **books** domain, we see that cross-domain methods do not achieve as good performance as the single-domain PersonalityMF, which only exploits target domain preferences and personality factors. In contrast, when dealing with the target domain of **movies**, we find that cross-domain information is indeed beneficial, most notably when personality factors are transferred from the books source domain. Likewise, in the **music** domain, we observe that cross-domain approaches clearly outperform PersonalityMF. In this case, exploit-

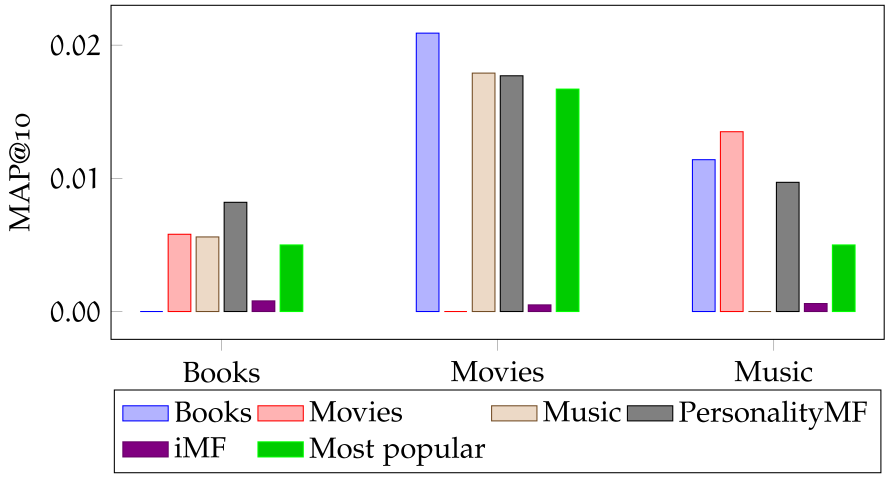

Figure 5.6: MAP@10 of cross-domain approaches in the extreme cold start scenario with no user overlap. The x axis represents the target domain.

ing personality factors from the music source domain yields the best recommendation accuracy. We note a remarkable asimmetry in the obtained results: transferring personality factors learned in the books domain is useful for improving movie and music recommendations. On the other hand, learning the effect of personality on movie and music likes is not helpful for predicting book likes.

Precision results in the moderate cold start without user overlap

The results for the *moderate* cold start are shown in Figure 5.7. We observe consistent trends through profile sizes from 1–10 likes. Therefore, we report the average MAP values for simplicity. As previously, we observe that cross-domain information is not valuable for **books** recommendations. Regarding **movies** and **music**, we see the same behaviour that in the single-domain scenario, i.e., exploiting personality information does not improve over iMF once target domain preferences are available. Moreover, cross-domain approaches do not outperform single-domain PersonalityMF, indicating that the transfer of personality factors from other domains does not provide any more information.

Novelty and coverage without user overlap

The results for *novelty* and *coverage* are presented in Table 5.4. For convenience, we include again the values corresponding to PersonalityMF in single domains. Regarding the extreme cold start, we observe that cross-domain methods can deliver more novel recommendations with greater coverage, depending on the source domain. For instance, we see that the cross-domain approach exploiting books information provides less popular recommendations than PersonaliyMF in the **movies** domain, while at the same time covering more items in the catalog as shown by the larger spread values. Similarly, in the **music** domain, exploiting cross-domain movie factors leads to more novel recommendations and greater coverage. In fact, we notice that cross-domain methods that perform better in terms of accuracy (see Fig-

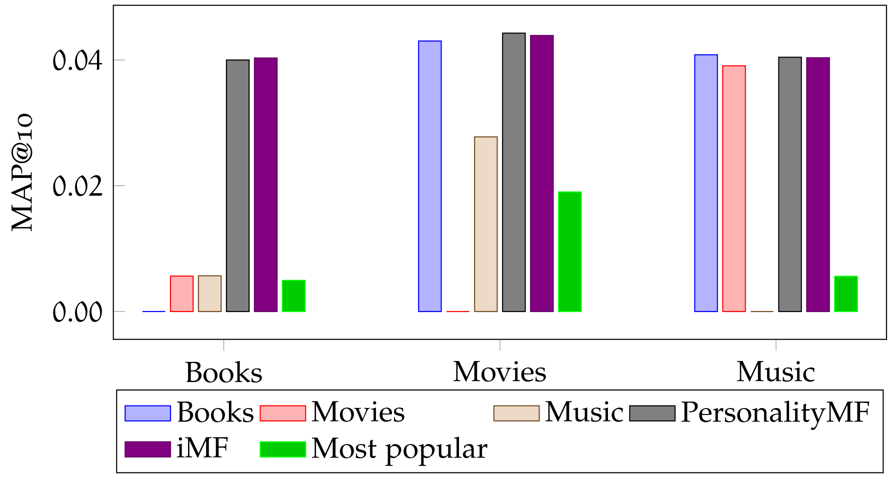

Figure 5.7: MAP@10 of cross-domain approaches in the moderate cold start scenario with no user overlap, averaged for profiles with 1–10 likes. The x axis represents the target domain.

ure 5.6) also achieve good results in terms of novelty and coverage. Nonetheless, in the moderate cold start we degrade novelty and coverage when exploiting cross-domain information, regardless of the considered source domain.

5.5.4 *Exploiting personality for cross-domain recommendation with user overlap between domains*

The goal of our fourth and last experiment is to understand whether personality information can be leveraged in addition to the users' source domain preferences in order to boost the performance of cross-domain recommendations. For such purpose, we compare the two approaches presented in Section 5.3.2, namely:

- The extended matrix factorization model that enhances user profiles with cross-domain preferences following Equation 5.7. We refer to this approach as `Books`, `Movies` or `Music`, depending on the source domain that is exploited.
- The matrix factorization model that further enhances user profiles with personality information *in addition to* cross-domain preferences, as in Equation 5.8. We refer to this model as `Books+Pers`, `Movies+Pers` or `Music+Pers`, again depending on the considered source domain. Note that these methods differ from those reported in Section 5.3.1 as they also exploit information from a source domain.

Precision results in extreme cold start

In Figure 5.8, we show the performance of the approaches in the *extreme* cold start scenario, for all the possible source-target domain configurations. In two cases out of three, combining personality information with cross-domain ratings further improves the performance

Table 5.4: Novelty and coverage of cross-domain approaches in the cold start. Values for the *moderate* scenario are averaged across profile sizes 1–10.

	Method	Extreme cold start		Moderate cold start	
		Avg. Pop.	Spread	Avg. Pop.	Spread
Books	Personality MF	185.19	5.26	144.59	6.27
	Movies	216.39	4.15	235.83	3.59
	Music	168.72	5.64	213.19	4.47
Movies	Personality MF	5717.94	4.75	4080.65	6.43
	Books	5085.92	5.38	4357.24	6.12
	Music	5822.97	4.32	5588.41	4.58
Music	Personality MF	9846.59	4.73	6592.58	6.86
	Books	9515.28	5.05	6712.47	6.78
	Movies	9298.19	5.14	7312.54	6.27

when no preferences about the user are available in the target domain. Only in the **books** domain, the best results are obtained using movie data only. In this case, adding personality information does not improve the recommendation performance, but it is beneficial if the available auxiliary information consists of music ratings (13.2% relative improvement over the cross-domain approach without personality). When predicting **movies** preferences, we observe that cross-domain approaches enhanced with personality information always achieve better recommendation performance. In fact, the overall best results are obtained by combining music preferences and personality (5% improvement of `Music+Pers` over `Music`), and if only book *likes* are available as auxiliary information, the accuracy can be further improved by considering personality (by 12.2%). In the case of **music** recommendations, we observe a symmetrical trend, where the best results are achieved combining personality with movie *likes* (16.7% improvement of `Music+Pers` over `Music`). On the other hand, adding book *likes* is clearly beneficial, but in this case exploiting personality information yields only a minimal improvement.

Precision results in moderate cold start

The results for the *moderate* cold-start are shown in Figure 5.9. Differently from the *extreme* cold-start scenario, we cannot conclude that personality is beneficial for larger user profile sizes in the **books** domain. In the case of **movies**, we obtain small improvements combining personality with music ratings, but the effect is the opposite when dealing with book ratings. Finally, when recommending **music**, we clearly see the advantages of combining personality with auxiliary movie ratings, which consistently gives the best overall results.

Novelty and coverage results

Regarding *novelty*, the average popularity of the recommended items remains roughly equal across the cross-domain approaches, but still

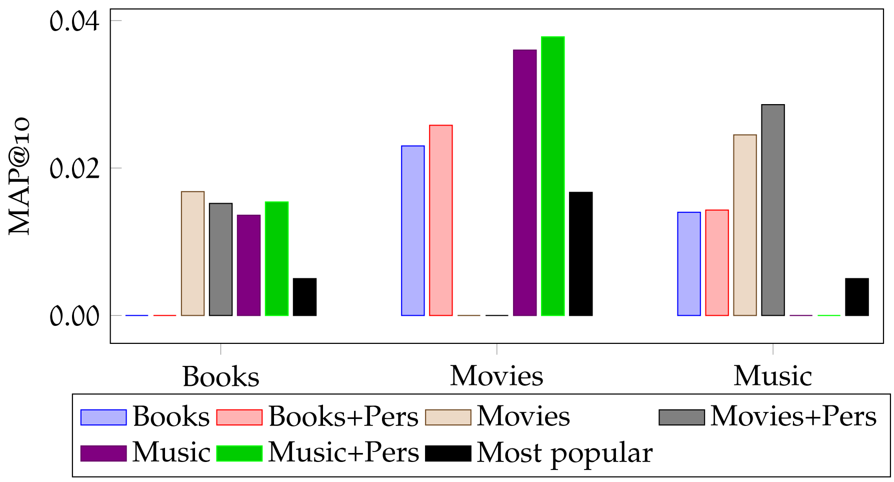

Figure 5.8: MAP@10 of cross-domain approaches with user overlap in the extreme cold start scenario. The x axis represents the target domain, and bars correspond to the approaches with different combinations of source domain with or without personality.

much lower than the Most popular baseline, as expected (on average, 145–148 vs. 236 in the books domain, 3,980–4089 vs. 6,620 for movies, and 6,622–6,753 vs. 11,092 for music). In terms of *coverage*, the spread of the item distribution is again similar among cross-domain methods, whereas it is much lower for the baseline (on average, 6.23–6.25 vs. 3.45 in books, 6.40–6.48 vs. 3.45 in movies, and 6.78–6.88 vs. 3.44 in music).

5.5.5 *Discussion*

In the experiments reported in this chapter we have seen that user personality information can be used individually in single- and cross-domain scenarios to improve the performance of a recommender system, especially for completely new users. We now compare the evaluated approaches against each other as solutions to the *extreme* cold start, as illustrated in Figure 5.1.

We intend to understand whether personality by itself is enough to provide good recommendations or if, in contrast, it is better to exploit cross-domain preferences in addition to personality. Additionally, we also evaluate the *active learning* baseline described in Section 5.4.2 as an alternative approach to the cold start, which elicits preferences directly from the user rather than exploiting auxiliary information.

In Figure 5.10 we compare the MAP@10 values of the best performing approaches for the new user problem in the *extreme* cold start situation, i.e., for users completely new to the system.

Personality-based cross-domain recommendation

Our **personality-based cross-domain matrix factorization** model is the best performing method in all the considered domains, effectively exploiting the additional source preferences. The boost in precision,

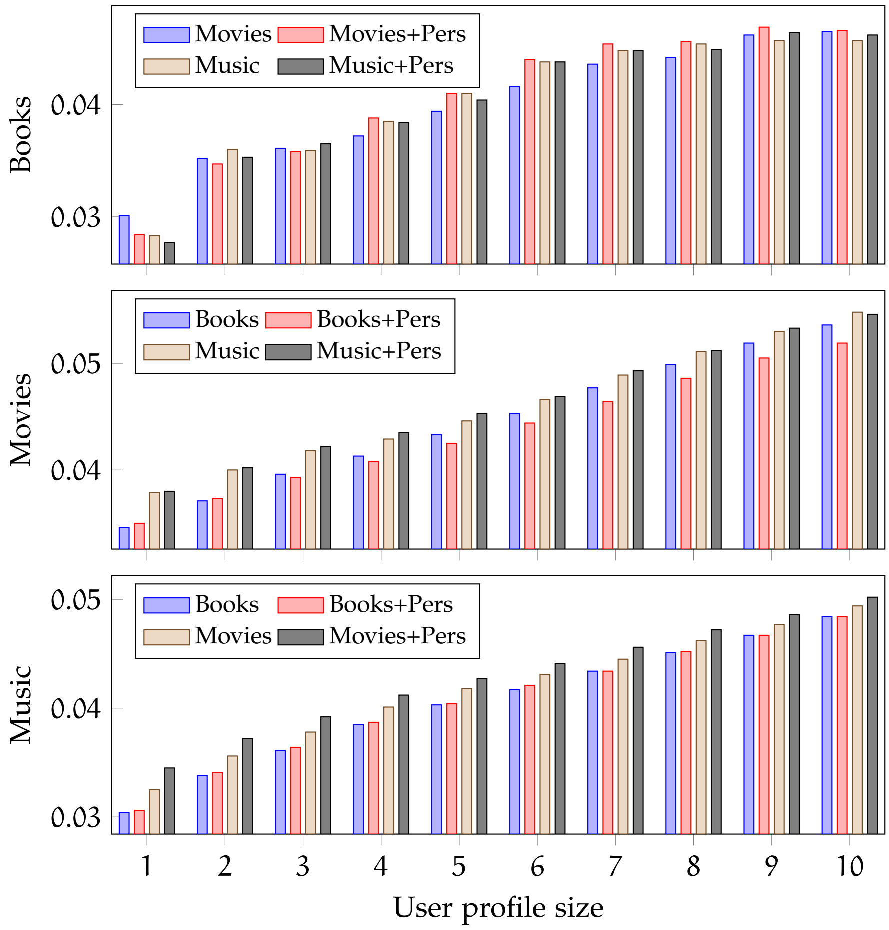

Figure 5.9: MAP@10 of cross-domain approaches with user overlap for different profile sizes.

specially in the movies and music domains, comes however at the cost of collecting the auxiliary information, and the time required to train the models. It can be a compelling approach if cross-domain preferences are available at the time of designing the target system — e.g., if the catalog of items is expanded with a new domain— and the goal is to optimize for precision regardless of the training complexity.

Personality-based single-domain recommendation

When no auxiliary preferences are available, the proposed **personality-based single-domain matrix factorization** model effectively exploits personality information when the cold start is extreme. Our approach is fast to train, and provides precision values better than the popularity baseline and than the iMF model by Hu et al., (2008), which is unable to compute meaningful recommendations in this scenario. In the moderate cold start situation, as more user preferences are available, we do not achieve significant improvements using personality with respect to iMF. We argue, nonetheless, that being able to provide recommendations for completely new users is a very desirable quality of recommender systems that is worth the acquisition of personality information.

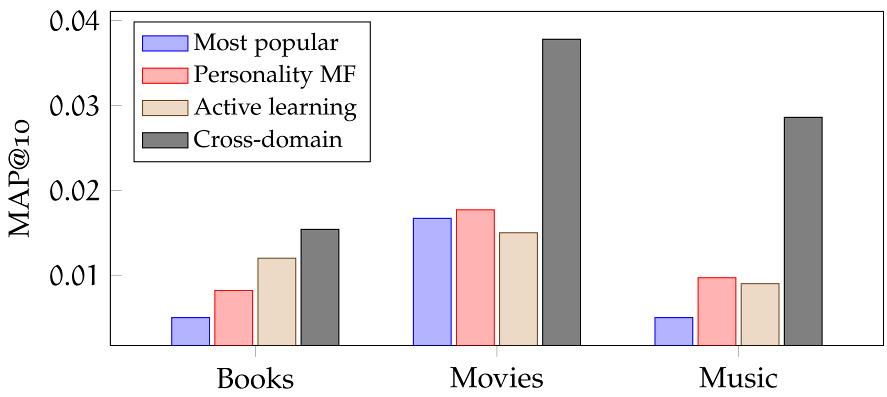

Figure 5.10: Comparison of evaluated approaches on the *extreme* cold start scenario.

Personality-based active learning

Finally, the **personality-based active learning** approach is a good alternative when the cold start is extreme and there are no auxiliary cross-domain preferences available, although it requires some effort from the user to provide an initial set of *likes*. We argue that additional aspects such as the design of the user interface are of great importance in this context. Although the recommendation model has to be trained again after user preference acquisition, the computational cost in this case is much lower than with cross-domain approaches. Also, the improvements in terms of precision are notable in the **books** and **music** domains. In the case of movies, we see that additional elicited *likes* are needed for the iMF baseline to achieve better performance, as users seem to favor popular movies. However, there is a clear trade-off between the effort required from the user and the gain in recommendation performance.

General results for each domain

Our findings are summarized in Table 5.5. In the **books** domain, we see that personality-based active learning is a compelling approach, as it offers good precision in the extreme cold start situation and overall good novelty and coverage. It is unclear if the boost in recommendation performance achieved by cross-domain approaches is worth the extra time required to train the models. For **movies**, personality-based cross-domain is clearly the best approach. It offers roughly twice the precision maintaining good novelty and coverage, and is also able to provide better performance as the number of available *likes* grows —using auxiliary music preferences, the best performing method. Finally, in the case of **music** recommendations, we find again that personality-based cross-domain is a compelling approach. However, due to the size of the dataset in this domain (see Table 5.1), the training time is considerably larger than for the other methods. Unless the extra precision is required, the single-domain matrix factorization model enhanced with personality is a good alternative, as it is the second best approach in terms of precision and offers better novelty and coverage than active learning in the extreme cold start.

Table 5.5: Summary of the performance of the different methods.

			Extreme cold start			Moderate cold start		
	Method	Time	Prec.	Nov.	Cov.	Prec.	Nov.	Cov.
Books	Most popular	✓						
	Personality MF	✓	✓	✓	✓		✓	✓
	Active learning	✓	✓	✓	✓		✓	✓
	Cross-domain		✓	✓	✓		✓	✓
Movies	Most popular	✓	✓					
	Personality MF	✓	✓	✓	✓	✓	✓	✓
	Active learning	✓			✓		✓	✓
	Cross-domain		✓	✓	✓	✓	✓	✓
Music	Most popular	✓						
	Personality MF	✓	✓	✓	✓		✓	✓
	Active learning	✓	✓		✓		✓	✓
	Cross-domain		✓	✓	✓	✓	✓	✓

5.6 CONCLUSIONS

Personality influences how people make decisions, and a number of studies have demonstrated the existence of correlations between personality traits and user preferences in multiple domains. Based on these findings, previous work has proposed adaptations of nearest neighbors heuristics for exploiting personality information in collaborative filtering, showing clear recommendation performance improvements.

In this chapter we have investigated the use of personality to support the transfer of knowledge in cross-domain recommendations for addressing the cold start. For that purpose, we have presented a novel approach that incorporates personality factors into a state of the art matrix factorization model with positive-only feedback. By jointly exploiting user preferences from a source domain and user personality information, our model is able to successfully recommend relevant items to new users. Moreover, the conducted experiments show that personality is beneficial even in single-domain recommendation, where source domain preferences are not available. Nonetheless, if source domain preferences are available, our findings indicate that cross-domain preferences enhanced with personality is the most successful approach to solve the cold start problem, outperforming active learning techniques that acquire target preference data directly from the users.

Beyond accuracy, our experiments show that, in this scenario, the personality-based models provide more novelty and coverage than the baselines. We have not reported results on recommendation diversity due to the lack of item content features needed to measure the pairwise similarities of items. However, in Chapter 6 we shall extract a similar dataset of Facebook likes linked to entities in the Semantic Web, which will allow us to obtain content-based features that will be used to compute recommendation diversity.

Notwithstanding the positive results for new users, the benefits of exploiting personality information practically vanish once some preferences are available in the target domain. Moreover, throughout this chapter we were not concerned with the acquisition of the users' personality information itself, and always assumed it was already available. Real-world systems must obtain this data from the users first by requesting to fill questionnaires, which users may not be interested in. In order to avoid this problem, recent work has explored the possibility of automatically inferring personality factors from the users' interactions with the system (Kosinski et al., 2013).

Another limitation of our cross-domain approach is the extra time needed to learn the matrix factorization models, as the number of variables grows with the amount of available source domain preferences, which can be very large, and can make the training process slow.

In summary, in this chapter we have presented a novel personality-based matrix factorization model that falls under the **knowledge transfer** category of cross-domain recommendation approaches that we described in Chapter 3. Like the other models proposed in this thesis, it aims to address the **linked-domain exploitation** cross-domain task with the goal of providing relevant recommendations for **cold start users**.

6 EXPLOITING ITEM METADATA IN MATRIX FACTORIZATION FOR CROSS-DOMAIN COLLABORATIVE FILTERING

In addition to collaborative filtering, content-based filtering has been applied in domains where item content and metadata play a key role, either in addition to or instead of explicit ratings and implicit user feedback. Examples of such domains are not limited to domains where items do have text contents —such as books, news articles, scientific papers, and web pages— and where text mining and information retrieval techniques are often used, but also domains where items have associated metadata (usually in the form of independent attributes), e.g., genres, directors and actors for movies, and music styles, composers and themes for songs. With the advent of the Semantic Web, and its reference implementation Linked Data, a plethora of structured, interlinked metadata is available on the Web. These metadata represent a potential source of information to be exploited by content-based and hybrid filtering approaches.

Hence, motivated by the use of Linked Data for recommendation purposes, in this chapter we present our last matrix factorization models for cross-domain collaborative filtering, which leverage metadata as a bridge between items liked by users in different domains. In Section 6.1 we motivate the use of item metadata for cross-domain recommendations, and introduce our approach. In Section 6.2 we review related works that exploit Linked Data as a source of item metadata for recommendation, and describe a state of the art algorithm to perform an efficient matrix factorization, which is the basis of the proposed recommendation approaches. In Section 6.3 we present our matrix factorization models, and in Section 6.4 we describe experiments conducted to evaluate the models. Finally, in Section 6.6 we provide some conclusions about the work done and the results achieved.

6.1 INTRODUCTION

To date the large majority of the proposed approaches to cross-domain recommendation deals with collaborative filtering, exploiting user preferences —usually expressed as explicit ratings for items— as a bridge to relate source and target domains, and ignoring any content-based description of the items. These approaches thus benefit from the fact that they do not need to perform any kind of analysis of item contents, which are in general highly heterogeneous across domains, and whose inter-relationships may be difficult to be established.

The Semantic Web and Linked Open Data

These difficulties, however, could be addressed nowadays thanks to the so called Semantic Web initiative (Shadbolt et al., 2006), and more specifically to its reference implementation the Linked Open Data (LOD) project (Bizer et al., 2009). While the former has allowed establishing standards for the use of common data formats and exchange protocols on the Web —such as the RDF[1] resource description format and the SPARQL[2] formal query language—, the latter has originated a large number of inter-linked knowledge repositories publicly available in the Web, following the Semantic Web standards for data representation and access. Hence, in the current Web there is a wide array of structured data sources with information of items belonging to a variety of domains, such as history, arts, science, industry, media and sports, to name a few. This information not only consists of particular multimedia contents and associated metadata, but also explicit, semantic relations between items and metadata.

Linked Data for content-based filtering

Motivated by the availability of large amounts of item metadata and semantic relations in the Linked Data cloud, in this chapter we aim to address the cross-domain recommendation problem in a flexible way: instead of focusing only on user/item rating information, we also propose to exploit content-based features and relations between items from different domains. The use of LOD does not merely allow describing items by means of (isolated) content-based features as done in the majority of CB approaches, but also establishing semantic networks that relate items, features, and items with features. For instance, Kubrick's *The Full Metal Jacket* is a movie based on Hasford's *The Short-Timers* novel, and *Anti-War Films* is a subgenre of *Political Films*. The set of LOD semantic features and relations could be exploited as inter-domain links for supporting knowledge transfer across domains, and this may let computing cross-domain item similarities and recommendations for cold start users in a target domain. Consider an example of a book and movie content-based recommender system, where a user has expressed some tastes for books, and expects suggestions of movies, domain in which she has not yet provided any preference. In this case, the film and literary genres could bridge such domains, identifying movies aligned with the user's preferred book genres as potential candidates for recommendation, e.g., suggesting *The Princess Bride* and *Willow* movies, if the user liked Tolkien's *The Lord of the Rings* and Lewis' *The Chronicles of Narnia* novels, since all these movies and books belong to the fantasy film/literary genre.

Social tags as non-intrinsic, non-linked content-based item features

In this context, it is worth noticing that a particular case of cross-domain content-based filtering approaches are those that exploit social tags. A user is characterized by the tags she assigned to the items she is interested in and, analogously, an item is represented by the

---

1 Resource Description Framework, RDF, `https://www.w3.org/TR/rdf11-primer`
2 SPARQL Query Language for RDF, `https://www.w3.org/TR/rdf-sparql-query`

set of tags the users have assigned to it. In Chapter 4 we showed that inter-domain relations established through social tags allow for an effective transfer of knowledge in cold start situations. In this chapter, in contrast, we focus on content-based features intrinsic to items, and not provided by the users. Moreover, as explained before, we assume that some of these features link items within and across domains, forming a heterogeneous, multi-domain knowledge graph, commonly referred to as *semantic network*.

Semantic networks for computing inter-domain item similarities

Previous work has proposed graph-based algorithms to address the recommendation problem in heterogeneous datasets (Di Noia et al., 2016; Kaminskas et al., 2013; Loizou, 2009; Yu et al., 2014), analyzing the topology of semantic networks to jointly exploit user preferences and item metadata. These approaches have been shown to be effective for recommendation, but suffer from computational issues caused by the size of the semantic networks, which are in general very large. Differently, the approaches presented in this chapter avoid these issues by working in two steps. First, they exploit the semantic networks to compute inter-domain similarities that link items from different domains. Then, they leverage the computed similarities in hybrid matrix factorization models for recommendation, which no longer need to deal with the whole semantic networks. In order to make more efficient the learning of our models, which not only have to learn the auxiliary source domain user preferences, but also the item metadata leveraged to bridge the domains, in this chapter we also present several adaptations of a fast training algorithm for matrix factorization proposed by Pilászy et al., (2010).

6.2 BACKGROUND AND RELATED WORK

In this section we first review related works that exploit Linked Data knowledge repositories for recommendation, in both single- and cross-domain scenarios. We then describe a fast, state of the art training algorithm upon which our models are built.

6.2.1 *Linked Open Data as a source of item metadata*

Lined Open Data

Within the Semantic Web initiative, the Linked (Open) Data[3] project aims to publish structured datasets —usually described by standard metadata models such as RDF— on the Web, and setting (RDF) links between data items —usually called semantic entities— from different structured data repositories —commonly referred to as knowledge bases. The adoption of Linked Data thus has led to the extension of the Web with a global data space connecting data from diverse domains such as people, companies, books, movies, television, music, statistical and scientific data, and reviews, to name a few.

3 The Linked Open Data project, http://linkeddata.org

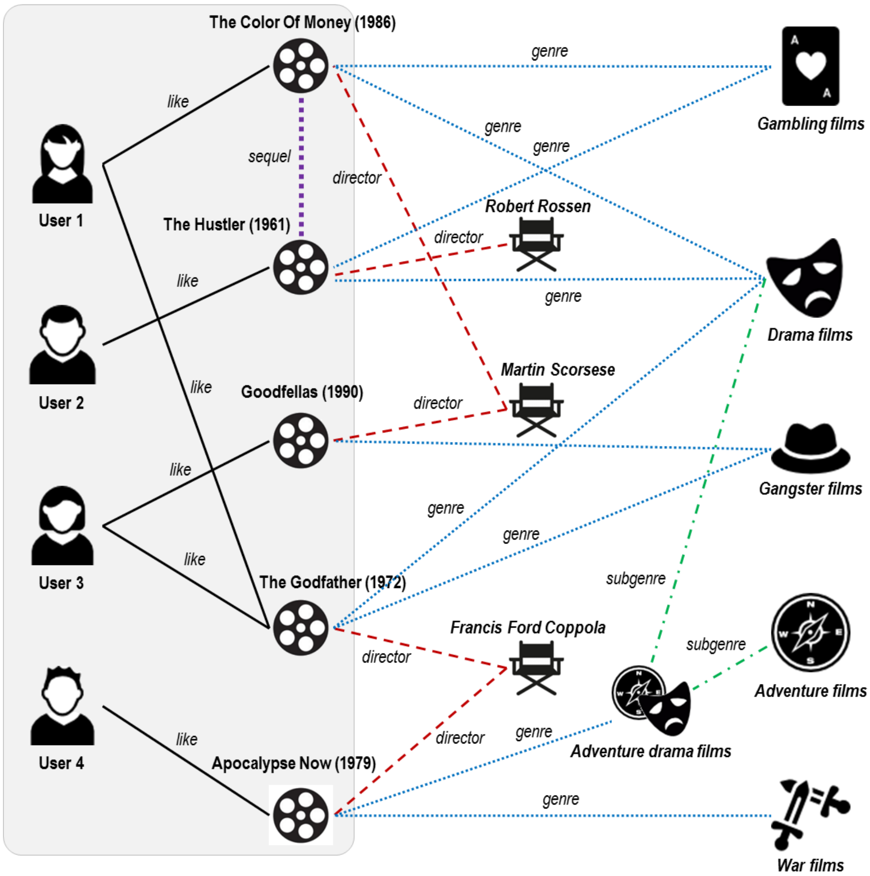

Figure 6.1: Example of heterogeneous information graph.

DBpedia Among the datasets existing in the Linked Data cloud, DBpedia (Lehmann et al., 2015) plays the role of a knowledge hub connecting many other data repositories. It is the LOD version of Wikipedia[4] and, as of November 2016, its knowledge base describes 4.58M entities, including 1.4M people, 735K places, 411K creative works, and 241K organizations. For each of them, DBpedia gathers metadata from structured data of the corresponding Wikipedia web pages. Such metadata are stored as RDF triples of the form `(subject, property, object)`, e.g., `(The Godfather, genre, Gangster films)` and `(Francis Ford Coppola, director, The Godfather)`, forming semantic networks of entities related through the properties in the triples.

Semantic networks Previous recommendation approaches have exploited these semantic networks directly. They have emerged concurrently with the increasing availability of additional user and item data useful for the recommendation process itself, combining the user-item rating matrix with side information into a graph, and then applying graph mining and ranking algorithms. As an illustrative example, Figure 6.1 shows the transformation of the rating matrix into a bipartite graph component (on the left of the figure) —consisting of user and item

4 The Wikipedia online encyclopedia, `http://www.wikipedia.org`

nodes linked by rating/*like* edges—, which is extended to form a multi-partite graph, including nodes that represent additional entities related to items. The graph also allows including other edges representing e.g. contextual information of the ratings, social connections between users, and semantic relations between entities (Shi et al., 2014). The result thus can be defined as a heterogeneous information graph consisting of a multi-typed and multi-relational directed graph, with nodes and edges of different nature.

Linked Data for single-domain graph-based recommendation

Aiming to exploit the heterogeneous semantic networks for recommendation, Passant, (2010) developed *dbrec*, a system built upon DBpedia that computes semantic distances between concepts to recommend related music bands and solo artists. More recently, Yu et al., (2014) presented *HeteRec*, a hybrid method that uses meta-path features to represent the connectivity between users and items along different types of paths in the network. HeteRec defines a user preference diffusion score extending the meta-path based similarity PathSim (Sun et al., 2011) to include implicit user feedback. This process propagates user preferences along the different meta-paths in the graph, producing a user-item matrix for each meta-path where each cell indicates the probability that a certain user reaches a certain item under the relative meta-path. Then, it factorizes each matrix, and builds a recommendation model that estimates the rating for a user-item pair computing a weighted sum of the relative meta-path features in the matrices. Finally, Di Noia et al., (2016) proposed *SPrank* (semantic path-based ranking), a hybrid algorithm able to combine ontological knowledge from LOD with collaborative user preferences in a unified graph-based data model in a learning to rank setting. Beyond item filtering, a recent work by Musto et al., (2016a) presented *ExpLOD*, a framework that exploits LOD to generate natural language explanations produced by recommendation algorithms. Aiming to reduce the size of the semantic networks, in (Musto et al., 2016b) the authors analyzed the impact of different LOD properties on recommendation performance, and applied feature selection techniques to automatically determine the most relevant features for recommendation.

Linked Data for cross-domain graph-based recommendation

As presented in Chapter 3, in the cross-domain recommender systems literature, there have been some attempts to establish semantic relations between items of different types. Loizou, (2009) proposed to identify explicit semantic relations between items, and exploit such relations for cross-domain recommendations. Specifically, items were annotated and linked by concepts and properties extracted from Wikipedia. Then, with such relations, users and items were incorporated into a graph, upon which a probabilistic recommendation model was built. In (Fernández-Tobías et al., 2011) we presented an approach that uses DBpedia as a multi-domain knowledge source for building a semantic network that links concepts from several domains. On such semantic network, which has the form of an acyclic directed graph, a weight spreading activation algorithm retrieves entities (items) in a target domain that are highly related to input entities and concepts in a source domain. Later, Kaminskas et al., (2014) applied the same idea to the task of recommending musicians and compositions related to places of interest (POIs) a user is visiting, showing that relevant items tend to be connected to the POIs through more paths in ad hoc cross-domain semantic network.

Linked Data for the proposed cross-domain matrix factorization models

In contrast to the above mentioned graph-based recommendation approaches, Rowe, (2014) presented *SemanticSVD++*, an extension of the well known SVD++ matrix factorization method (Koren, 2008). The proposed approach includes additional parameters to capture the evolution of the users' tastes on semantic categories, which is shown to provide more accurate rating predictions. Similarly to the previous works, in this chapter we use Semantic Web technologies and Linked Data repositories to establish relations between items from different domains. However, instead of following a graph-based ranking algorithm, which can be very costly when the used semantic network is large, we propose three matrix factorization models that exploit item metadata extracted from LOD. Differently from (Rowe, 2014), our models are based on inter-domain item semantic similarities for regularization, rather than adding new latent parameters, which greatly increase the computational cost of the algorithm. Moreover, our models are designed for dealing with positive-only feedback in the item ranking task, as opposed to numerical rating predictions.

6.2.2 *Fast Alternating Least Squares-based matrix factorization*

Fast Alternating Least Squares

In this section we describe an algorithm proposed by Pilászy et al., (2010) for fast learning in matrix factorization. The algorithm is based on *Alternating Least Squares* (ALS), and reduces the computational complexity of the matrix factorization models of Hu et al., (2008). The recommendation models presented in this chapter make use of this optimization and remain effective while incorporating the extra cross-domain user preferences and item attribute metadata.

The ALS-based approach to matrix factorization that we reviewed in Section 2.3.2 works by iteratively fixing a set of parameters —user and item factors, respectively—, and optimizing the remainder parameters by analytically solving a k-dimensional least squares problem (see Equation 2.25 and Equation 2.26). The major computational bottleneck in this approach is the calculation of a matrix inverse for each user and item, with a total cost of $\mathcal{O}(k^2|\mathcal{R}_+| + k^3|\mathcal{U}|)$ and $\mathcal{O}(k^2|\mathcal{R}_+| + k^3|\mathcal{I}|)$ for each step, where $\mathcal{R}_+$ is the set of observed user-item interactions.

Rather than finding an exact solution for the k-dimensional least squares problem, the approach by Pilászy et al., (2010) computes an approximation by solving k distinct univariate least squares problems, one respectively for each coordinate, while keeping the others fixed. Let $\{\vec{x}_n\}_{n=1}^N$ be a collection of N examples that represent the independent variable, $\{y_n\}_{n=1}^N$ their respective observed outputs, and $\{c_n\}_{n=1}^N$ the confidence placed on each observation. The goal of ordinary least squares is to find a vector $\vec{w} \in \mathbb{R}^k$ such that $y_n \approx \langle \vec{w}, \vec{x}_n \rangle$, for every $1 \leqslant n \leqslant N$. In practice, the optimal $\vec{w}$ is found by minimizing the squared error of each example as follows:

$$\mathcal{L}(\vec{w}) = \sum_{i=1}^{N} c_n \left(y_n - \langle \vec{w}, \vec{x}_n \rangle\right)^2 + \lambda \left\|\vec{w}\right\|^2 \tag{6.1}$$

where the regularization parameter λ prevents overfitting by favoring sparse solutions. This loss function is convex, and thus has a global minimum, which can be obtained analytically by setting $\frac{\partial \mathcal{L}}{\partial \vec{w}} = 0$ and solving for $\vec{w}$, leading to:

$$\vec{w} = \left(\mathbf{X}^\top \mathbf{C} \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{C} \vec{y} \tag{6.2}$$

where the matrix $\mathbf{X} \in \mathbb{R}^{N \times k}$ contains the $\vec{x}_n$ as rows, $\mathbf{C} \in \mathbb{R}^{N \times N}$ is a diagonal matrix with the c_n values, and $\vec{y} \in \mathbb{R}^N$ is a column vector with all the y_n's. Computing $\vec{w}$ requires inverting a matrix of size $k \times k$ and the product $\mathbf{X}^\top \mathbf{C} \mathbf{X}$, which results in a computational complexity of $\mathcal{O}(k^3 + k^2 N)$. In ALS-based matrix factorization for recommendation, a least squares problem must be solved for each user and item, resulting in very expensive computations, specially when N is large, as is the case in the cross-domain setting.

The RR1 algorithm

The approximation by Pilászy et al., (2010) optimizes one component of $\vec{w}$ at a time, while keeping the rest fixed. Let w_α be the α-th component of $\vec{w}$ that we aim to optimize, and fix w_β for every $\alpha \neq \beta$. The goal of the resulting univariate least squares problem is to find w_α such that $e_n \triangleq y_n - \sum_{\beta \neq \alpha} w_\beta x_{n\beta} \approx w_\alpha x_{n\alpha}, \forall 1 \leqslant n \leqslant N$. This can be achieved by minimizing the component-specific loss function:

$$\mathcal{L}_\alpha(w_\alpha) = \sum_{n=1}^{N} c_n \left(e_n - w_\alpha x_{n\alpha}\right)^2 + \lambda w_\alpha^2 \tag{6.3}$$

As previously, the optimal w_α is found by setting $\frac{d\mathcal{L}}{dw_\alpha} = 0$:

$$w_\alpha = \frac{\sum_{i=1}^{N} c_n e_n x_{n\alpha}}{\sum_{i=1}^{N} c_n x_{n\alpha}^2 + \lambda} \tag{6.4}$$

This solution has computational complexity $\mathcal{O}(N)$, and is computed for each coordinate $1 \leqslant \alpha \leqslant k$, resulting in a total running time of $\mathcal{O}(kN)$.

The above algorithm for approximate least squares solutions, referred to as RR1, is then applied to ALS-based matrix factorization for collaborative filtering (see Algorithm 1) as follows. In the **P-step**, all the item parameters $\vec{q}_i$ are fixed, and an optimal latent vector $\vec{p}_u$ is found for each user by minimizing Equation 2.24:

$$\mathcal{L}_u(\vec{p}_u) = \sum_i c_{ui} \left(r_{ui} - \langle \vec{p}_u, \vec{q}_i \rangle\right)^2 + \lambda \left\|\vec{p}_u\right\|^2 + \text{constant} \tag{6.5}$$

The constant includes the terms in Equation 2.24 that do not depend on u. The previous formula describes a multivariate least squares problem just like Equation 6.1. Hence, we apply RR1 to Equation 6.5 for each user with the following settings:

- The training examples correspond to the fixed item parameters, $\vec{x}_i := \vec{q}_i$, $\forall i \in \mathcal{I}$, and the total number of examples is $N := |\mathcal{I}|$.
- Observed user preferences play the role of the dependent variables, $y_i := r_{ui}$.
- Confidence values are specific to the current user, $c_i := c_{ui}$.
- The parameter to optimize is $\vec{w} := \vec{p}_u$.

The total complexity of the P-step for all users is therefore $\mathcal{O}(k|\mathcal{U}||\mathcal{I}|)$.

Similarly, in the **Q-step**, the user parameters $\vec{p}_u$ are fixed, and the optimal item factors $\vec{q}_i$ are chosen to minimize:

$$\mathcal{L}_i(\vec{q}_i) = \sum_u c_{ui} \left(r_{ui} - \langle \vec{p}_u, \vec{q}_i \rangle\right)^2 + \lambda \left\|\vec{q}_i\right\|^2 + \text{constant} \tag{6.6}$$

Again, we apply RR1 to each item using the following values:

- The training examples correspond to the fixed user parameters, $\vec{x}_u := \vec{p}_u$, $\forall u \in \mathcal{U}$, and the total number of examples is $N := |\mathcal{U}|$.
- Observed user preferences play the role of the dependent variables, $y_u := r_{ui}$.
- Confidence values are specific to the current item, $c_u := c_{ui}$.
- The parameter to optimize is $\vec{w} := \vec{q}_i$.

The computational complexity for all items is again $\mathcal{O}(k|\mathcal{U}||\mathcal{I}|)$.

In (Pilászy et al., 2010) the authors provide further optimizations that we do not describe here as they are out of the scope of this thesis. We refer the reader to that work, and only mention that such optimizations allow the complexity to drop to $\mathcal{O}(k^2|\mathcal{U}| + k^2|\mathcal{I}| + k|\mathcal{R}|)$, resulting in a very efficient training algorithm. Rather than analyzing the computational complexity of the training algorithms, the experiments performed in this chapter focus on the evaluation of cross-domain recommendation models for the cold start. We notice here, however, that testing the RR1 approach on our datasets, we observed training times up to $10\times$ faster than using the traditional ALS from Hu et al., (2008).

6.3 proposed recommendation models

In this section we present our three matrix factorization cross-domain recommendation models for positive-only feedback that exploit item metadata to bridge the source and target domains. The models presented in Chapter 4 and Chapter 5 extended MF by including new sets of parameters to model social tags and personality factors, respectively. Instead of adding new parameters, in this case we leverage the auxiliary information in the form of item metadata to compute similarities between items from different domains, which are in turn used to regularize the learned parameters.

In this context, items from different domains tend to have very diverse attributes that are not straightforward related. For instance, a book may be characterized by its *author* or by its *book genres*, and a movie can be described using its *cast*, *director* or *movie genres*. In fact, content-based features are often different between domains, and even when they refer to related concepts, such as *book genres* and *movie genres*, the features may not be directly aligned, e.g., *funny movies* vs. *comedy books*.

Computing inter-domain item semantic similarities

In order to overcome the heterogeneity of features of items from different domains, we propose to exploit Linked Data for linking entities from multiple and diverse domains. Specifically, we map the items in our datasets to entities in DBpedia. In Section 6.4.1 we shall describe the process of mapping items to semantic entities from DBpedia.

Once the items are mapped to their corresponding entities, we use the DBpedia graph to compute semantic similarities between such entities, mining both the attributes and the structure of the graph with semantic relations. More specifically, we exploit the information in DBpedia to compute a semantic similarity matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{I}_S| \times |\mathcal{I}_T|}$ between the source domain items $\mathcal{I}_S$ and the target domain items $\mathcal{I}_T$:

$$s_{ij} = \text{sim}(i, j), \qquad i \in \mathcal{I}_S, j \in \mathcal{I}_T \tag{6.7}$$

In Section 6.5 we shall report recommendation performance results by using several semantic similarity metrics from the state of the art.

Incorporating inter-domain item semantic similarities into matrix factorization

The computed inter-domain item similarities are then used to *link* the domains for cross-domain recommendation (see Section 3.5.1). In the cold start, when a user has rated a few (if any) items in the target domain, a recommender system could suggest the user with items in the target domain that are semantically similar to those the user liked in the source domain. Hence, the system could be effective only if there is an overlap of users between the domains. Moreover, even cold start users in the target domain should have some preferences in the source domain.

In the next subsections we present our three recommendation models based on the exploitation of semantic similarities to regularize item factors in MF, so that similar items from different domains tend to have similar parameters. In this way, even if the user's preferences in the target domain are unknown, a recommender system could suggest the user with target items that are most similar to those she preferred in the source.

6.3.1 *Regularization through similarity prediction*

Joint factorization of rating and inter-domain similarity matrices

The first semantic-based matrix factorization cross-domain model we propose is based on the assumption that latent vectors of related items should explain the items semantic similarities, in addition to the users' preferences. That is, we not only seek to predict the preferences $r_{ui} \approx \langle \vec{p}_u, \vec{q}_i \rangle$, but also the inter-domain similarities $s_{ij} \approx \langle \vec{q}_i, \vec{q}_j \rangle$, where $i \in \mathcal{I}_S$ and $j \in \mathcal{I}_T$.

Hence, our model jointly factorizes the rating and inter-domain item similarity matrices that link the source and target domains. Let $\mathcal{U} = \mathcal{U}_S \cup \mathcal{U}_T$ be the set of all users, which we assume overlaps between the domains, and let $\mathcal{I} = \mathcal{I}_S \cup \mathcal{I}_T$ be the set of all items, which we assume do not overlap. Our model learns a latent vector $\vec{p}_u \in \mathbb{R}^k$ for each user $u \in \mathcal{U}$, but separately models source and target domain items $\vec{q}_i$ and $\vec{q}_j$, with $i \in \mathcal{I}_S$ and $j \in \mathcal{I}_T$, as follows:

$$\mathcal{L}(\mathbf{P}, \mathbf{Q}_S, \mathbf{Q}_T) = \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{I}} c_{ua} \left(r_{ua} - \langle \vec{p}_u, \vec{q}_a \rangle \right)^2 + \lambda_C \sum_{i \in \mathcal{I}_S} \sum_{j \in \mathcal{I}_T} \left(s_{ij} - \langle \vec{q}_i, \vec{q}_j \rangle \right)^2 + \lambda \left(\|\mathbf{P}\|^2 + \|\mathbf{Q}_S\|^2 + \|\mathbf{Q}_T\|^2 \right) \tag{6.8}$$

where $\mathbf{Q}_S$ and $\mathbf{Q}_T$ are matrices containing the item latent vectors as rows from the source and target domains, respectively. We note that the summation in the first term iterates over all items $a \in \mathcal{I}$ from both domains, as we want to factorize the source and target user-item preference matrices simultaneously. The cross-domain regularization parameter $\lambda_C > 0$ controls the contribution of the inter-domain semantic similarities; large values of the parameter will force items to have too similar latent vectors, whereas low values will result in limited transfer of knowledge between domains.

Alternating Least Squares

As in standard matrix factorization, we train our model using Alternating Least Squares. First, we fix $\mathbf{Q}_S$ and $\mathbf{Q}_T$, and solve analytically for each $\vec{p}_u$ by setting the gradient to zero. Since the user factors do not appear in the additional cross-domain regularization term, we obtain the same solution as for the baseline MF model (see Equation 2.25):

$$\vec{p}_u = \left(\mathbf{Q}^\top \mathbf{C}^u \mathbf{Q} + \lambda \mathbf{I} \right)^{-1} \mathbf{Q}^\top \mathbf{C}^u \vec{r}_u \tag{6.9}$$

In order to simplify the notation, we have defined the matrix $\mathbf{Q}$ as the row-wise concatenation of $\mathbf{Q}_S$ and $\mathbf{Q}_T$. The matrix $\mathbf{C}^u$ is a diagonal matrix with the confidence values c_{ua} for all $a \in \mathcal{I}$, and the vector $\vec{x}_u$ contains the preferences of user u, again for all items $a \in \mathcal{I}$.

Next, we fix the user factors $\mathbf{P}$ and the target domain item factors $\mathbf{Q}_T$, and compute the optimal values for the source domain item factors. Again, by setting the corresponding gradient to zero and solving analytically we obtain:

$$\vec{q}_i = \left(\mathbf{P}^\top \mathbf{C}^i \mathbf{P} + \lambda_C \mathbf{Q}_T^\top \mathbf{Q}_T + \lambda \mathbf{I}\right)^{-1} \left(\mathbf{P}^\top \mathbf{C}^i \vec{r}_i + \lambda_C \mathbf{Q}_T^\top \vec{s}_i\right) \quad (6.10)$$

As previously, the vector $\vec{r}_i$ contains the preferences assigned to item i, and $\vec{s}_i$ is the i-th row of the inter-domain semantic similarity matrix $\mathbf{S}$. Finally, we proceed as before fixing $\mathbf{P}$ and $\mathbf{Q}_S$ to compute the optimal solution for the target domain item latent vectors:

$$\vec{q}_j = \left(\mathbf{P}^\top \mathbf{C}^j \mathbf{P} + \lambda_C \mathbf{Q}_S^\top \mathbf{Q}_S + \lambda \mathbf{I}\right)^{-1} \left(\mathbf{P}^\top \mathbf{C}^j \vec{r}_j + \lambda_C \mathbf{Q}_S^\top \vec{s}_j\right) \quad (6.11)$$

RR1 for fast Alternating Least Squares

The computation of the optimal factors can be parallelized within each step, but the larger number of items to consider and the extra step required for the source domain greatly increase the training time with respect to the MF baseline. In order to address this issue, we adapt the fast training algorithm for ALS described in Section 6.2.2. Since the computation of the user factors is the same as in the original MF model, the procedure remains the same for the P-step. For the source domain Q-step, by inspecting Equation 6.8 and Equation 6.10, we note that the additional terms that arise from the inter-domain similarities can be treated just like user preferences as follows. For each source item i:

1. Generate examples for each rating r_{ui} as for baseline MF (see Section 6.2.2)

2. For each target item $j \in \mathcal{I}_T$:
   - Generate an input example $\vec{x}_j := \vec{q}_j$.
   - Use the similarity as the dependent variable, $y_j := s_{ij}$.
   - Use a constant confidence value $c_j := \lambda_C$.
   - The parameter to optimize is $\vec{w} := \vec{q}_j$.

The above procedure will produce in the similarity terms of Equation 6.10, which can be defined by means of the confidence matrix $\tilde{\mathbf{C}}^i = \lambda_C \mathbf{I}$. The procedure for the source domain Q-step is completely analogous.

6.3.2 *Regularization based on item neighborhoods*

Neighborhood inter-domain similarities to regularize item latent factors

Our second semantic-based matrix factorization cross-domain model exploits the item semantic similarities in a different fashion. Instead of forcing pairwise item interactions to reproduce the observed similarity values, the approach we present here leverages $\mathbf{S}$ to regularize the item latent vectors, so that feature vectors of similar items are pushed together in the latent space. Intuitively, items that are semantically similar should also have similar latent parameters.

As previously, let $\mathcal{U} = \mathcal{U}_S \cup \mathcal{U}_T$ and $\mathcal{I} = \mathcal{I}_S \cup \mathcal{I}_T$ be the sets of all users and items, respectively. Our approach jointly factorizes the source and target domain rating matrices, and regularizes similar item factors proportionally to the items similarity. However, instead of considering all the potentially similar source domain items, we limit the regularization of a target domain item $j \in \mathcal{I}_T$ to its neighborhood, i.e., to the set $N(j) \subseteq \mathcal{I}_S$ of the top-n most similar source domain items:

$$\begin{aligned}\mathcal{L}(\mathbf{P}, \mathbf{Q}_S, \mathbf{Q}_T) = &\sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{I}} c_{ua} \left(r_{ua} - \langle \vec{p}_u, \vec{q}_a \rangle\right)^2 \\ &+ \lambda_C \sum_{j \in \mathcal{I}_T} \sum_{i \in N(j)} s_{ij} \left\| \vec{q}_j - \vec{q}_i \right\|^2 + \lambda \left(\sum_{u \in \mathcal{U}} \|\vec{p}_u\|^2 + \sum_{a \in \mathcal{I}} \|\vec{q}_a\|^2 \right)\end{aligned} \tag{6.12}$$

We note that items with greater similarity values are more heavily regularized, whereas items with values of $s_{ij} \approx 0$ in their neighborhoods are barely affected. However, it may still be convenient to regularize such items so that they benefit from cross-domain information, and thus may be eligible for recommendation to cold start users. Therefore, we also experiment normalizing the similarity scores in the item neighborhoods so that $\sum_{i \in N(j)} s_{ij} = 1$. In this way all target items are equally regularized, but each is affected by its source domain neighbors proportionally to their similarity scores.

By assigning latent vectors to target domain items close to those of similar source domain items, our model is able to generate recommendations in cold start settings. Specifically, let $\vec{q}_j$ be the latent vector learned for target item $j \in \mathcal{I}_T$, and let $\vec{q}_i$ be the latent vector of source item $i \in \mathcal{I}_S$, which we assume is semantically similar to j. Our model will regularize both factors so that their distance $\left\|\vec{q}_j - \vec{q}_i\right\|$ is small, or equivalently, $\vec{q}_j \approx \vec{q}_i$. Consider now a cold start user u who only provided preferences in the source domain, so that her corresponding latent vector $\vec{p}_u$ is therefore only adjusted using source domain preferences. In standard MF, it is not guaranteed that $\vec{p}_u$ will extrapolate to the target domain, and will provide an accurate prediction for $\vec{q}_j$. In contrast, our model ensures that $\langle \vec{p}_u, \vec{q}_j \rangle \approx \langle \vec{p}_u, \vec{q}_i \rangle$, i.e., target domain items yield relevance prediction scores close to that of similar source domain items. Hence, u will be recommended with a target domain item j if the user liked the source domain item i, or if i would be recommended to u in the source domain.

Once more, we train our neighborhood-based matrix factorization model using Alternating Least Squares. As in the previous model, the user factors are not affected by the extra regularization, and can be computed again using Equation 6.9, leaving the P-step unchanged. For the target domain item factors $\vec{q}_j$ we proceed as usual, fixing the user and source item factors, and finding the values such that $\frac{\partial \mathcal{L}}{\partial \vec{q}_j} = 0$, which yields the solution:

Alternating Least Squares

$$\vec{q}_j = \left[\mathbf{P}^\top \mathbf{C}^j \mathbf{P} + \left(\lambda + \lambda_C \sum_{i \in N(j)} s_{ij} \right) \mathbf{I} \right]^{-1} \left(\mathbf{P}^\top \mathbf{C}^j \vec{r}_j + \lambda_C \sum_{i \in N(j)} s_{ij} \vec{q}_i \right) \tag{6.13}$$

Repeating the same procedure for the source item factors $\vec{q}_i$ we obtain:

$$\vec{q}_i = \left[\mathbf{P}^\top \mathbf{C}^i \mathbf{P} + \left(\lambda + \lambda_C \sum_{j \in N^{-1}(i)} s_{ij} \right) \mathbf{I} \right]^{-1} \left(\mathbf{P}^\top \mathbf{C}^i \vec{r}_i + \lambda_C \sum_{j \in N^{-1}(j)} s_{ij} \vec{q}_j \right) \tag{6.14}$$

where $N^{-1}(i)$ is the *inverse neighborhood* of item i, i.e., the set of target domain items that have i among their neighbors: $N^{-1}(i) = \{j \in \mathcal{I}_T | i \in N(j)\}$.

Unlike the model presented in the previous section, we cannot apply RR1 directly by treating the new similarity terms as additional user preferences. Instead, we derive again the update rules for each component of the source and target domain item parameters. As mentioned before, user parameters remain unchanged. Let $j \in \mathcal{I}_T$ be a target item, and consider the optimization of the α-th component $q_{j\alpha}$ of its corresponding latent vector $\vec{q}_j$. We can rewrite the loss in Equation 6.12 as a function only of $q_{j\alpha}$ as follows:

Optimization of Alternate Least Squares

$$\begin{aligned} \mathcal{L}_\alpha(q_{j\alpha}) = &\sum_{u \in \mathcal{U}} c_{uj} \left(e_{uj} - p_{u\alpha} q_{j\alpha} \right)^2 + \lambda q_{j\alpha}^2 \\ &+ \lambda_C \sum_{i \in N(j)} s_{ij} \left(q_{j\alpha} - q_{i\alpha} \right)^2 + \text{constant} \end{aligned} \tag{6.15}$$

where $e_{uj} \triangleq r_{uj} - \sum_{\beta \neq \alpha} p_{u\beta} q_{j\beta}$, and the constant includes terms that do not depend on $q_{j\alpha}$. If we set the derivative $\frac{d\mathcal{L}_\alpha}{dq_{j\alpha}} = 0$, we obtain:

$$q_{j\alpha} = \frac{\sum_{u \in \mathcal{U}} c_{uj} e_{uj} p_{u\alpha} + \lambda_C \sum_{i \in N(j)} s_{ij} q_{i\alpha}}{\sum_{u \in \mathcal{U}} c_{uj} p_{u\alpha}^2 + \lambda + \lambda_C \sum_{i \in N(j)} s_{ij}} \tag{6.16}$$

Using the optimizations described in (Pilászy et al., 2010), the computational cost of the above formula for all items is $\mathcal{O}(k^2|\mathcal{U}| + k|\mathcal{R}| +$

$n|\mathcal{I}_T|$), since all the neighborhoods are formed using the top n most similar items, $|N(j)| \leqslant n$. Applying the same procedure to the source domain item factor $\vec{q}_i$ we obtain:

$$q_{i\alpha} = \frac{\sum_{u \in \mathcal{U}} c_{ui} e_{ui} p_{u\alpha} + \lambda_C \sum_{j \in N^{-1}(i)} s_{ij} q_{j\alpha}}{\sum_{u \in \mathcal{U}} c_{ui} p_{u\alpha}^2 + \lambda + \lambda_C \sum_{j \in N^{-1}(i)} s_{ij}} \tag{6.17}$$

The main difference with respect to Equation 6.16 is that the sets $N^{-1}(i)$ are not bounded, as a source item can potentially be the neighbor of an arbitrary number of target items, so that $|N^{-1}(i)| \leqslant |\mathcal{I}_T|$, resulting in a theoretical worst-case cost of $\mathcal{O}(k^2|\mathcal{U}| + k|\mathcal{R}| + |\mathcal{I}_S||\mathcal{I}_T|)$. We observe, however, that in practice most of the source items appear only in a few neighborhoods and that the algorithm is still very efficient.

6.3.3 *Regularization based on item centroids*

Neighborhood centroid inter-domain similarities to regularize item latent factors

When neighbor source domain items are mutually diverse, the neighborhood-based model presented in the previous section may struggle to regularize a target domain item that has to be simultaneously close to all its neighbors. The model we propose in this section works like the neighborhood-based model, but, instead of using the neighbor source domain items *individually* in the regularization, it uses their *centroid* (average) latent vector:

$$\begin{aligned} \mathcal{L}(\mathbf{P}, \mathbf{Q}_S, \mathbf{Q}_T) = & \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{I}} c_{ua} \left(r_{ua} - \langle \vec{p}_u, \vec{q}_a \rangle \right)^2 \\ & + \lambda_C \sum_{j \in \mathcal{I}_T} \left\| \vec{q}_j - \sum_{i \in N(j)} s_{ij} \vec{q}_i \right\|^2 + \lambda \left(\sum_{u \in \mathcal{U}} \|\vec{p}_u\|^2 + \sum_{a \in \mathcal{I}} \|\vec{q}_a\|^2 \right) \end{aligned} \tag{6.18}$$

The same considerations regarding the neighborhood $N(j)$ and the normalization of the similarity scores also apply to this model. However, the effect on the item relevance predictions for cold start users is different. Let $\vec{q}_j$ be an item in the target domain, and let $N(j)$ be its neighborhood of most similar source domain items. The regularization scheme in our centroid-based approach aims to minimize the distance $\left\| \vec{q}_j - \sum_{i \in N(j)} s_{ij} \vec{q}_i \right\|$, so that the latent vector of item j is close, *on average*, to those of the source items in $N(j)$, i.e., $\vec{q}_j \approx \sum_{i \in N(j)} s_{ij} \vec{q}_i$. Let u be a cold start user in the target domain that has some preferences in the source domain. Again, her feature vector $\vec{p}_u$ is only learned using the user's source preferences, and may not be reliable for computing relevance predictions for target domain items in standard MF. Our model, however, ensures that

$$\langle \vec{p}_u, \vec{q}_j \rangle \approx \left\langle \vec{p}_u, \sum_{i \in N(j)} s_{ij} \vec{q}_i \right\rangle = \sum_{i \in N(j)} s_{ij} \langle \vec{p}_u, \vec{q}_i \rangle$$

That is, the predicted relevance score is roughly the average of the relevance scores for the neighbor source domain items, weighted by their corresponding semantic similarity.

As in the previous models, the user parameters are not affected by the item regularization terms, and can be computed in the standard fashion using Equation 6.9. For the target domain item factors $\vec{q}_j, j \in \mathcal{I}_T$, we set the gradient of Equation 6.18 to zero to obtain:

$$\vec{q}_j = \left[\mathbf{P}^\top \mathbf{C}^j \mathbf{P} + (\lambda + \lambda_C)\,\mathbf{I}\right]^{-1} \left(\mathbf{P}^\top \mathbf{C}^j \vec{r}_j + \lambda_C \sum_{i \in N(j)} s_{ij}\vec{q}_i\right) \quad (6.19)$$

Comparing the above to Equation 6.13 we observe that both are equivalent when $\sum_{i \in N(j)} s_{ij} = 1$, i.e., normalizing the similarity values has the same effect of than centroid-based regularization on the target domain item factors. The solution for source item factors $\vec{q}_i$, in contrast, has a different form:

$$\vec{q}_i = \left[\mathbf{P}^\top \mathbf{C}^i \mathbf{P} + \left(\lambda + \lambda_C \sum_{j \in N^{-1}(i)} s_{ij}^2\right)\mathbf{I}\right]^{-1} \cdot \left(\mathbf{P}^\top \mathbf{C}^i \vec{r}_i + \lambda_C \sum_{j \in N^{-1}(j)} s_{ij}\left(\vec{q}_j - \vec{z}_{j\setminus i}\right)\right) \quad (6.20)$$

where we have defined $\vec{z}_{j\setminus i} = \sum_{l \in N^{-1}(i), l \neq i} s_{lj}\vec{q}_l$ to simplify the notation. We note that, differently to the previous models presented in this chapter, the computation of the source domain latent vectors cannot be parallelized, as the value of $\vec{q}_i, i \in \mathcal{I}_S$ depends on the values of other $\vec{q}_l, l \in \mathcal{I}_S$ through the parameter $\vec{z}_{j\setminus i}$. As a result, the training process can be slow when the set of source domain items is large. In our experiments, however, we observed that the time penalty of computing the source factors sequentially is usually compensated by the faster RR1 algorithm, although we do not provide any quantitative analysis as it falls out of the scope of this work.

RR1 for fast Alternate Least Squares

In order to apply RR1 to our centroid-based approach, we derive again the solutions for each α-th coordinate separately. Once more, the solution for the user factors remains the same as it is not affected by the regularization terms. For the target domain item factors $\vec{q}_j$, we consider the loss in Equation 6.18 as a function only of the α-th component $q_{j\alpha}$:

$$\mathcal{L}_\alpha(q_{j\alpha}) = \sum_{u \in \mathcal{U}} c_{uj}\left(e_{uj} - p_{u\alpha}q_{j\alpha}\right)^2 + \lambda q_{j\alpha}^2 + \lambda_C \left(q_{j\alpha} - \sum_{i \in N(j)} s_{ij}q_{i\alpha}\right)^2 + \text{constant} \quad (6.21)$$

As previously, the constant includes terms that do not depend on $q_{j\alpha}$, and e_{uj} is defined as in Equation 6.15. Setting the derivative $\frac{d\mathcal{L}_\alpha}{dq_{j\alpha}} = 0$ yields:

$$q_{j\alpha} = \frac{\sum_{u\in\mathcal{U}} c_{uj} e_{uj} p_{u\alpha} + \lambda_C \sum_{i\in N(j)} s_{ij} q_{i\alpha}}{\sum_{u\in\mathcal{U}} c_{uj} p_{u\alpha}^2 + \lambda + \lambda_C} \tag{6.22}$$

We note, once again, the similar form of the above solution with respect to the previous model in Equation 6.16. If we apply the same procedure to the source domain item factors, we obtain:

$$q_{i\alpha} = \frac{\sum_{u\in\mathcal{U}} c_{ui} e_{ui} p_{u\alpha} + \lambda_C \sum_{j\in N^{-1}(i)} s_{ij} (q_{j\alpha} - \vec{z}_{(j\backslash i)\alpha})}{\sum_{u\in\mathcal{U}} c_{ui} p_{u\alpha}^2 + \lambda + \lambda_C \sum_{j\in N^{-1}(i)} s_{ij}^2} \tag{6.23}$$

The computational complexity for the target domain factors is equivalent to the model from the previous section, whereas for the source domain factors it is $\mathcal{O}(k^2|\mathcal{U}| + k|\mathcal{R}| + n|\mathcal{I}_S||\mathcal{I}_T|)$ in the worst case, which is similar to the neighborhood-based model since the size of the neighborhoods n is in general small.

6.4 EXPERIMENTS

6.4.1 *Dataset*

Facebook likes for items from multiple domains

Our dataset initially consisted of a large set of *likes* assigned by users to items in Facebook. Using the Facebook Graph API, a user's *like* is retrieved in the form of a 4-tuple with the following information: the identifier, name and category of the liked item, and the timestamp of the like creation, e.g., `{id: "35481394342", name: "The Godfather", category: "Movie", created_time: "2015-05-14T12:35:08+0000"}`. The name of an item is given by the user who created the Facebook page of such item. In this context, distinct names may exist for a particular item, e.g., *The Godfather, The Godfather: The Movie, The Godfather - Film series*, etc. Users thus may express likes for different Facebook pages which actually refer to the same item. Aiming to unify and consolidate the items extracted from Facebook likes, we developed a method that automatically maps the items names with the unique URIs of the corresponding DBpedia entities, e.g., http://dbpedia.org/resource/The_Godfather for the identified names of *The Godfather* movie.

Mapping item names to DBpedia entity labels

LINKING ITEMS TO DBPEDIA ENTITIES Given a particular item, we first identified DBpedia entities that are labeled with the name of the item. For such purpose, we launched a SPARQL query targeted on the subjects of triples that have `rdfs:label` [5] as property and the item title as object. The next query is an example for *The Matrix 2* title:

5 Namespace for rdfs, http://www.w3.org/2000/01/rdf-schema

```
SELECT DISTINCT ?item WHERE {
  {
    ?item rdf:type dbo:Film .
    ?item rdfs:label ?name .
    FILTER regex(?name, "the.*matrix.*2", "i") .
  }
  UNION
  {
    ?item rdf:type dbo:Film .
    ?tmp dbo:wikiPageRedirects ?item .
    ?tmp rdfs:label ?name .
    FILTER regex(?name, "the.*matrix.*2", "i") .
  }
}
```

To resolve ambiguities in those names that correspond to multiple items belonging to different domains, we specify the type of the item we wanted to retrieve in each case. Specifically, the previous query includes a triple clause with `rdf:type`[6] (or `dbo:type`[7]) as property. Hence, in the given example, the subject `The Matrix 2` refers to the "movie" type, which is associated to the `dbo:Film` class in DBpedia. The item types were set from the item categories provided in Facebook, and their associated DBpedia and YAGO[8] classes[9] were identified by manual inspection of the `rdf:type` values of several entities. Table 6.1 shows the list of item types and DBpedia/YAGO classes we considered for the three domains of our dataset.

Moreover, running the previous query template we observed that a number of items were not linked to DBpedia entities because the labels corresponded to Wikipedia *redirection* webpages. In these cases, to reach the appropriate entities the query makes use of the `dbo:wikiPageRedirects` property. The result of the previous query for *The Matrix 2* is `http://dbpedia.org/resource/The_Matrix_Reloaded`, which actually is the DBpedia entity of the second movie in *The Matrix* saga. Here, it is important to note that thanks to the Wikipedia page redirect component we were able to link items whose names do not have a direct syntactic match with the label of its DBpedia entity, but with the label of a redirected entity, e.g., the *Matrix 2* title matches the `The Matrix Reloaded` entity.

FINAL SEMANTICALLY ANNOTATED DATASET For every linked entity, we finally accessed DBpedia to retrieve the metadata that afterward will be used as input for the recommendation models. In this case, we launched a SPARQL query asking for all the properties and

Retrieving DBpedia entity metadata

6 Namespace for rdf, `http://www.w3.org/1999/02/22-rdf-syntax-ns#`
7 Namespace for dbo, `http://dbpedia.org/ontology`
8 The YAGO knowledge base, `http://www.mpi-inf.mpg.de/yago-naga/yago`
9 Namespace for yago, `http://dbpedia.org/class/yago`

Table 6.1: Considered item types and their DBpedia and YAGO classes for the three domains of the dataset.

	Item type	DBpedia/YAGO classes
Books	Book	dbo:Book, yago:Book102870092, yago:Book102870526
	Genre	yago:LiteraryGenres
	Writer	dbo:Writer, yago:Writer110794014
	Fictional character	dbo:FictionalCharacter, yago:FictionalCharacter109587565
Movies	Movie	dbo:Film, yago:Movie106613686
	Genre	dbo:MovieGenre, yago:FilmGenres
	Director	yago:FilmDirector110088200, yago:Director110014939
	Actor	dbo:Actor, yago:Actor109765278
	Fictional character	dbo:FictionalCharacter, yago:FictionalCharacter109587565
Music	Composition	dbo:Song, dbo:MusicalWork, dbo:Single, dbo:ClassicalMusicComposition, dbo:Opera
	Genre	dbo:MusicGenre, yago:MusicGenres, yago:MusicGenre107071942
	Album	dbo:Album, yago:Album106591815
	Musician	dbo:MusicalArtist, yago:Musician110339966, yago:Musician110340312, yago:Composer109947232
	Band	dbo:Band, yago:MusicalOrganization108246613

objects of the triples that have the target entity as subject. Following the example given before, such a query would be:

```
SELECT ?p ?o WHERE {
  dbr:The_Matrix_Reloaded ?p ?o .
}
```

This query returns all the DBpedia property-value pairs of the `dbr:The_Matrix_Reloaded`[10] entity. However, since our ultimate goal is item recommendation, we should only exploit metadata that may be relevant to relate common preferences of different users. Thus, we filtered the query results by considering certain properties in each domain. Specifically, Table 6.2 shows the list of DBpedia properties selected for each of the three domains of our dataset. Hence, for example, for the movie items, we would have as metadata the movies genres, directors, and actors, among others.

The items and relations shown in the table thus represent a *semantic network* that is automatically obtained from DBpedia for each particular domain. Table 6.3 shows statistics of the dataset for the three domains of interest, namely books, movies, and music.

Three types of item metadata

semantically enriched item profiles Fixing *books*, *movies*, *musicians* and *bands* as the target items to be recommended, we can distinguish the following three types of item metadata obtained:

- *attributes*, which correspond to item-attribute entities associated to the considered item types of Table 6.2, and are distinct to the entities of target items, e.g., the genre(s), director(s) and actors of a particular movie.
- *related items*, which correspond to the item-item properties in Table 6.2 that derive related entities, e.g., the novel a movie is based on (`dbo:basedOn` property), the prequel/sequel of a movie (`dbo:previousWork`/`dbo:subsequentWork` properties), or the musicians belonging to a band (`dbo:bandMember` property).
- *extended attributes*, which correspond to attribute-attribute properties that generate extended item attributes, originally not appearing as metadata, e.g., the subgenres of a particular music genre (`dbo:musicSubgenre` property).

The above three types of item metadata constitute the semantically enriched item profiles that we propose to use in our recommendation models. We note that they differ from the commonly used content-based item profiles composed of plain attributes. We also note that in the conducted experiments, the results achieved by exploiting the enriched profiles were better than those achieved by only using item attributes.

10 Namespace for dbr, http://dbpedia.org/resource

Table 6.2: DBpedia properties considered as item metadata; *item* can be book, movie and composition, musician and band.

Relation	**DBpedia properties**
item – genre	`dct:subject, dbo:genre`
book – genre	`dbo:literaryGenre`
music genre – music genre	`dbo:musicSubgenre, dbo:musicFusionGenre, dbo:movement, dbo:derivative, dbo:stylisticOrigin`
item – author	`dbo:author, dbo:creator`
book – writer	`dbo:writer`
movie – actor, character, director	`dbo:starring, dbo:cinematograpy, dbo:director`
composition – musician	`dbo:artist, dbo:composer, dbo:musicComposer, dbo:musicalArtist, dbo:associatedMusicalArtist`
music item – album	`dbo:album`
band – musician	`dbo:bandMember, dbo:formerBandMember, dbo:musicalBand, dbo:associatedBand`
item – item, character	`dbo:series`
item – character	`dbo:portrayer`
item – item	`dbo:basedOn, dbo:previousWork, dbo:subsequentWork, dbo:notableWork`

Table 6.3: Statistics of the extracted dataset enriched with metadata.

	Books	**Movies**	**Music**
Users	1876	26 943	49 369
Items	3557	3901	5748
Likes	42 869	876 501	2 084 462
Sparsity (%)	99.4	99.2	99.3
Avg. items/user	22.85	32.53	42.22
Avg. users/item	12.05	224.69	362.64

6.4.2 *Evaluated approaches*

Evaluated recommendation approaches

We evaluated the following recommendation algorithms in the cross-domain scenario:

- **POP**. Non personalized baseline that always recommends the most popular items not yet liked by the user. Popularity is measured as the number of users in the dataset that liked the item.
- **UNN**. User-based nearest neighbors with Jaccard similarity. The size of the neighborhood is tuned for each dataset using a validation set.
- **INN**. Item-based nearest neighbors with Jaccard similarity and indefinite neighborhood size.
- **iMF**. Matrix factorization method for positive-only feedback (Hu et al., 2008) trained using the fast ALS technique by Pilászy et al., (2010).
- **HeteRec**. Graph-based recommender system proposed in (Yu et al., 2014), based on a diffusion method of user preferences following different meta-paths.
- **SimMF**. Our matrix factorization model regularized with similarity prediction proposed in Section 6.3.1.
- **NeighborMF**. Our proposed matrix factorization model with neighborhood-based regularization from Section 6.3.2.
- **CentroidMF**. Our matrix factorization model from Section 6.3.3 that uses the neighbor's centroid to regularize the target domain item factors.

For UNN, INN, IMF and HeteRec we considered their application to both single- and cross-domain scenarios. Hereafter we use the prefix `CD-` to indicate that the single-domain algorithm is using the union of the rating matrices from the source and target domains. We did not consider for our evaluation the SemanticSVD++ method by Rowe, (2014), as it is designed for rating prediction rather than item ranking. Moreover, preliminary tests showed that its performance was much lower than the other methods, and that its training time was about one order of magnitude larger.

Parameter setting

We tuned the hyperparameters of the considered recommendation models using a held-out validation set of likes, as we explain in the next section. For UNN, we only had to select the size of the user neighborhoods. For the matrix factorization models, in contrast, the number of hyperparameters is larger, namely, the dimensionality of the latent factor space k, the amount of regularization λ, and the confidence parameter for positive-only feedback α. Moreover, the models proposed in this chapter also include the cross-domain regularization rate λ_C, which controls the contribution of the inter-domain item similarities. Finally, for NeighborMF and CentroidMF, we tuned the size n of the item neighborhoods $N(j)$, and the possibility to normalize the neighbors' similarities so that the sum to 1, as explained in Section 6.3.2.

The high number of parameters to tune rules out the possibility of performing a grid search for the best values. Hence, we used Bayesian Optimization techniques (Snoek et al., 2012) that train Machine Learning models to predict candidate values that are likely to maximize a given function while simultaneously reducing the uncertainty of over unknown parameter values.

We tuned the parameters of the single-domain methods UNN and iMF only on the target domain, and used the same values for their cross-domain variants CD-UNN and CD-iMF. For UNN, the optimal number of neighbors was $n = 50$ for books, and $n = 100$ for movies and music. For iMF we obtained the optimal parameters $k = (10, 29, 21), \lambda = (10^{-5}, 0.823, 1)$, and $\alpha = (6, 7, 10)$ for books, movies, and music, respectively. The optimal values for our proposed cross-domain models are reported in Table 6.4.

6.4.3 *Evaluation methodology and metrics*

Evaluation methodology and metrics

In the conducted experiments, we followed the cold start evaluation methodology proposed by Kluver and Konstan, (2014) that we extended for the cross-domain recommendation setting in Section 5.4.3. The only difference with respect to the experiments reported in Chapter 5 is that the validation set is now used to tune the models, as explained in the previous section, rather than for user preference acquisition by *active learning* strategies.

Regarding the metrics, we used the Mean Reciprocal Rank (MRR) to evaluate the ranking accuracy of the recommendations, which computes the average reciprocal rank of the first relevant item in the recommendation list. Binomial Diversity Framework (BinomDiv) (Vargas et al., 2014) was used to evaluate the individual diversity, namely the degree of diversity in the recommendation lists based on item genres extracted from DBpedia.

6.5 RESULTS

In this section we present the results of the conducted experiments to evaluate the proposed matrix factorization models. First, we analyze several semantic relatedness metrics to compute the inter-domain item similarities. Next, we report the ranking accuracy and diversity of the evaluated recommendation approaches, and study how the size

Table 6.4: Optimal hyperparameters for SimMF, NeighborMF, and CentroidMF. The last column indicates whether the similarities in the neighborhood are normalized or not.

	Source	**Method**	k	λ	α	λ_C	n	**Norm.**
Books	Movies	SimMF	112	0	1	10^{-8}		
		NeighborMF	134	1	1	9.125	49	✓
		CentroidMF	153	0.999	1	8.778	100	✓
	Music	SimMF	10	1	16	10^{-8}		
		NeighborMF	10	0	18	10	100	✓
		CentroidMF	10	0	14	0.109	100	
Movies	Books	SimMF	12	1	1	0.002		
		NeighborMF	12	1	1	10	81	✓
		CentroidMF	14	0.100	1	0.200	1	✓
	Music	SimMF	35	0	1	1.6×10^{-6}		
		NeighborMF	51	1	1	10	100	
		CentroidMF	29	1	1	9.494	99	✓
Music	Books	SimMF	10	1	1	0.039		
		NeighborMF	10	0.995	1	3.014	100	✓
		CentroidMF	10	0.724	1	1.673	14	
	Movies	SimMF	11	0.571	4	0.641		
		NeighborMF	10	0.978	2	0.699	46	
		CentroidMF	10	0.562	2	10	3	✓

and diversity of the source domain user profile impacts on the target recommendations.

6.5.1 *Inter-domain item semantic similarity*

Approaches to inter-domain semantic relatedness

The goal of our first experiment is to analyze the performance of several semantic relatedness metrics to compute the inter-domain similarities that we later exploit in our matrix factorization models. We considered the following strategies:

- TF-IDF. We use the semantically-enriched item profiles (see Section 6.4.1 to build TF-IDF vector profiles based on the metadata of each item. The similarity score between a source domain item and a target domain item is computed as the cosine of their corresponding TF-IDF vectors.
- ESA. The Explicit Semantic Analysis technique proposed by Gabrilovich and Markovitch, (2007). Instead of using the semantic metadata, we map each item to its corresponding Wikipedia article. Then, based on the text of the article, ESA extracts a set of other related Wikipedia articles, which represent semantic concepts, and builds a TF-IDF profile from the extracted concepts. Finally, the similarity score between two items is computed as the cosine of their corresponding concept-based vectors.
- M&W. The approach proposed by Milne and Witten, (2008) computes the semantic relatedness between two items using the overlap of their sets of inlinks and outlinks in the Wikipedia hyperlink graph.
- Katz. Based on Katz's centrality measure, the relatedness between two items is computed as the accumulated probability of the top shortest paths between their corresponding entities in the semantic network (Hulpus et al., 2015).

Selecting a semantic relatedness metric for computing inter-domain item similarities

We evaluated the previous semantic relatedness metrics indirectly by comparing their performance in the item recommendation task. For such purpose, we chose a content-based recommendation model with no parameters, so that we can fairly measure the effect of each similarity on the item ranking quality. According to this simple model, the relevance score of an item is computed as the accumulated similarity with the items in the user's profile:

$$s(u,i) = \sum_{j \in I(u)} s_{ij} \tag{6.24}$$

where s_{ij} is computed any of the methods described above.

Table 6.5: MRR of the evaluated semantic relatedness metrics.

Source	Target	TF-IDF	ESA	M&W	Katz
Books	Movies	0.058	0.030	**0.123**	0.092
	Music	0.028	0.015	**0.042**	0.022
Movies	Books	**0.054**	0.011	0.031	0.013
	Music	**0.030**	0.011	0.028	0.009
Music	Books	0.010	0.006	**0.052**	0.020
	Movies	0.013	0.018	**0.088**	0.006

The results of our experiment are shown in Table 6.5. For easier comparison according the methodology from Section 6.4.3, we averaged the MRR scores for all the cold start sizes in each source-target domain combination. We conclude from the table that M&W is the best performing metric, beating all the other approaches except when considering the movie domain as source, in which case it is still competitive. Hence, in the following experiments we evaluate our proposed matrix factorization models using M&W as the backing semantic similarity. Finally, we note that the low values for MRR are due to the simple recommendation algorithm chosen for this experiment.

6.5.2 *Item ranking accuracy*

In our second experiment we analyze the accuracy of the item rankings generated by the evaluated recommendation approaches. We aim to understand if cross-domain variants are in general more effective than single-domain ones, and whether the proposed matrix factorization models are able to outperform the other methods in cold start settings.

Book recommendations

Table 6.6 shows the ranking accuracy for book recommendations in terms of MRR. We report the average results for cold start user profiles from sizes 6–10, as we observed that in those cases the trends are stable and, in general, single-domain baselines start to be effective. We notice from the table that, with the exception of UNN, any approach exploiting cross-domain movies or music preferences is able to provide better recommendations than the POP baseline. In case auxiliary movie preferences are available, we observe that the proposed NeighborMF and CentroidMF models achieve the best performance when only 1–3 book likes are observed. Moreover, in that case, our cross-domain matrix factorization models perform much better than the single-domain baselines. However, once 4 likes are available, CD-INN and single-domain HeteRec are more effective approaches. When the auxiliary preferences consist of music likes, we see that

Table 6.6: Accuracy (MRR) for cold start users in the books domain. Best values for each single- and cross-domain configuration are shown in bold.

		Number of book *likes*						
	Method	0	1	2	3	4	5	6–10
	POP	**0.242**	**0.244**	0.246	0.248	0.251	0.252	0.260
	UNN		0.222	**0.265**	**0.286**	0.289	0.290	0.322
	INN		0.145	0.177	0.216	0.241	0.262	0.316
	iMF		0.171	0.194	0.235	0.255	0.271	0.301
	HeteRec		0.218	0.244	0.279	**0.297**	**0.316**	**0.351**
Movies	CD-UNN	0.186	0.148	0.170	0.175	0.189	0.190	0.212
Movies	CD-INN	0.262	0.265	0.275	0.291	**0.301**	**0.307**	**0.339**
Movies	CD-iMF	0.261	0.262	0.268	0.272	0.275	0.274	0.287
Movies	CD-HeteRec	**0.264**	0.248	0.261	0.268	0.278	0.277	0.298
Movies	SimMF	0.253	0.268	0.274	0.284	0.289	0.290	0.296
Movies	NeighborMF	0.253	**0.272**	0.282	**0.294**	0.293	0.293	0.301
Movies	CentroidMF	0.252	0.271	**0.283**	0.289	0.293	0.295	0.301
Music	CD-UNN	0.136	0.103	0.115	0.120	0.138	0.140	0.157
Music	CD-INN	0.259	0.260	**0.266**	**0.278**	**0.296**	**0.302**	**0.329**
Music	CD-iMF	0.259	**0.261**	0.262	0.264	0.266	0.270	0.282
Music	CD-HeteRec	**0.266**	0.249	0.251	0.259	0.270	0.267	0.281
Music	SimMF	0.255	0.259	0.258	0.264	0.268	0.273	0.281
Music	NeighborMF	0.253	0.258	0.258	0.263	0.267	0.273	0.280
Music	CentroidMF	0.255	0.259	0.260	0.264	0.267	0.273	0.281

Table 6.7: Accuracy (MRR) for cold start users in the movies domain.

		Number of movie *likes*						
	Method	0	1	2	3	4	5	6–10
	POP	**0.285**	0.287	0.289	0.292	0.294	0.297	0.292
	UNN		**0.332**	0.320	0.318	0.330	0.348	0.330
	INN		0.233	0.300	0.336	0.359	**0.377**	0.321
	iMF		0.256	0.291	0.314	0.334	0.348	0.309
	HeteRec		0.315	**0.346**	**0.357**	**0.366**	0.374	**0.352**
Books	CD-UNN	0.219	0.169	0.185	0.219	0.256	0.292	0.224
Books	CD-INN	0.344	**0.347**	**0.371**	**0.386**	**0.398**	**0.410**	**0.382**
Books	CD-iMF	0.267	0.298	0.325	0.347	0.365	0.377	0.342
Books	CD-HeteRec	**0.479**	0.320	0.349	0.359	0.367	0.375	0.354
Books	SimMF	0.328	0.334	0.348	0.361	0.371	0.382	0.359
Books	NeighborMF	0.330	0.335	0.348	0.361	0.371	0.383	0.360
Books	CentroidMF	0.329	0.332	0.346	0.359	0.371	0.378	0.357
Music	CD-UNN	**0.387**	0.282	0.305	0.320	0.334	0.348	0.318
Music	CD-INN	0.342	0.347	0.353	0.359	0.365	0.371	0.359
Music	CD-iMF	0.301	0.326	0.344	0.362	0.374	0.385	0.358
Music	CD-HeteRec	0.367	0.336	0.344	0.350	0.355	0.360	0.349
Music	SimMF	0.339	0.351	0.361	0.374	0.384	0.396	0.373
Music	NeighborMF	0.353	**0.364**	**0.374**	**0.385**	**0.394**	**0.404**	**0.384**
Music	CentroidMF	0.345	0.355	0.367	0.377	0.385	0.395	0.376

CD-INN is the overall best method, although it is only useful for profiles of size 1. For larger profiles, it is better to use single-domain baselines than any cross-domain method that uses music preferences. In summary, we conclude that music preferences are not useful for book recommendations, whereas movie likes could be used to improve the performance, specially with NeighborMF and CentroidMF for 1–3 book likes.

Movie recommendations

In Table 6.7 we show the results for movie recommendations. We observe that most of the cross-domain approaches are able to provide recommendations better than the most popular items for completely new movie users, and that CD-HeteRec is clearly the best performing approach. If the auxiliary cross-domain data consists of book preferences, we notice that the proposed matrix factorization models outperform the best single-domain baselines. However, in this situation CD-INN is even a better method, clearly providing more accurate recommendations than any other approach from profile sizes 1–10. This is due to the high degree of overlap between the users of books and movies domains, which allows CD-INN to compute very accu-

Table 6.8: Accuracy (MRR) for cold start users in the music domain.

	Method	**Number of music *likes*** 0	1	2	3	4	5	6–10
	POP	**0.335**	0.337	0.340	0.342	0.345	0.347	0.342
	UNN		**0.422**	0.389	0.389	0.419	0.448	0.413
	INN		0.320	0.391	0.426	**0.455**	**0.474**	0.413
	iMF		0.347	**0.396**	**0.427**	0.451	0.471	**0.418**
	HeteRec		0.358	0.395	0.421	0.442	0.463	0.416
Books	CD-UNN	0.290	0.244	0.266	0.300	0.344	0.387	0.308
Books	CD-INN	0.310	0.368	**0.416**	**0.442**	**0.465**	**0.482**	**0.435**
Books	CD-iMF	0.200	0.330	0.391	0.423	0.451	0.471	0.413
Books	CD-HeteRec	**0.514**	0.367	0.407	0.432	0.453	0.474	0.427
Books	SimMF	0.310	0.368	0.401	0.424	0.446	0.461	0.420
Books	NeighborMF	0.328	**0.372**	0.402	0.425	0.445	0.461	0.421
Books	CentroidMF	0.325	0.370	0.402	0.425	0.444	0.461	0.420
Movies	CD-UNN	0.435	0.274	0.306	0.336	0.369	0.400	0.337
Movies	CD-INN	0.412	**0.431**	**0.451**	**0.467**	**0.478**	**0.490**	**0.463**
Movies	CD-iMF	0.293	0.356	0.398	0.428	0.454	0.474	0.422
Movies	CD-HeteRec	**0.515**	0.406	0.426	0.442	0.451	0.464	0.438
Movies	SimMF	0.361	0.393	0.420	0.438	0.455	0.467	0.435
Movies	NeighborMF	0.353	0.385	0.409	0.429	0.445	0.458	0.425
Movies	CentroidMF	0.354	0.386	0.413	0.431	0.447	0.460	0.428

rate item similarities based on the patterns of likes. Instead, when the source domain contains music preferences, we see that NeighborMF, CentroidMF, and SimMF, in that order, are consistently the best performing approaches for sizes 1–10. By regularizing item factors independently, NeighborMF is able to transfer source domain knowledge more effectively, which we also note is due to the greater contribution of cross-domain information (larger values of λ_C in Table 6.4). In summary, both book and music preferences are helpful for cold start movie recommendations, while our models are more effective when exploiting auxiliary music likes.

Music recommendations

Finally, the results for music recommendations are shown in Table 6.8. As previously, CD-HeteRec is a very good performing approach to provide recommendations for completely new users, in both cross-domain configurations. Once 2 music likes are available, CD-INN is clearly the most competitive approach, independently of the used source domain. Again, we argue that this is due to the high number of music users who also have book and movie preferences, which allows CD-INN to compute very accurate rating-based similarities for items. However, when the source domain consists of book preferences, we see that the proposed NeighborMF and CentroidMF models are slightly better than other cross-domain approaches if only 1 music like is provided. Anyway, even better performance can be achieved in this case simply using the single-domain UNN baseline, which does not need any extra information. Hence, single-domain baselines are compelling approaches for cold start music recommendations, and even though the proposed models are able to improve the quality of the item rankings by exploiting cross-domain item metadata, CD-INN, which is purely based on patterns of likes, is the best performing approach.

6.5.3 *Recommendation diversity*

In this subsection we analyze the diversity of the recommendation lists generated by the methods, as an alternative dimension of ranking quality.

Diversity of book recommendations

Table 6.9 shows the diversity of book recommendations in terms of the Binomial Diversity metric at cutoff 10 (BinomDiv@10). We observe that, in general, cross-domain approaches provide more diverse recommendations than their single-domain counterparts. However, we note several differences with respect to the accuracy results reported in Table 6.6. First, CD-UNN is consistently the superior algorithm in terms of diversity, whereas its accuracy results were the poorest among single- and cross-domain approaches. Second, when the source domain consists of movie likes, our proposed models achieve slightly worse diversity than other cross-domain approaches, specially for book profile sizes between 1–3 likes. This is in contrast with the results obtained in Table 6.6, where our methods performed best precisely in that range. We conclude that there is a clear trade-off between recommendation accuracy and diversity, and that the metric of interest depends on the particular application domain. We argue, however, that in cold start situations providing relevant suggestions may be more useful than recommending diverse, but not relevant items, if the ultimate goal of a system is to keep new users engaged.

Diversity of movie recommendations

The diversity results for movie recommendations are summarized in Table 6.10. As in the previous case, we notice that CD-UNN provides the most diverse but not relevant recommendations. Comparing the sources of auxiliary user preferences, we note that the diversity of the cross-domain baselines is roughly the same as their single-domain versions (comparing e.g. HeteRec and CD-HeteRec) when considering book likes. If the source domain contains music likes, in contrast, their diversity is significantly hurt. Finally, we remark the good performance of the NeighborMF method when source music likes are exploited, as it is able to provide a good trade-off of decent diversity and the most accurate recommendations (see Table 6.7).

Table 6.9: Diversity (BinomDiv@10) for cold start users in the books domain.

	Method	Number of book *likes* 0	1	2	3	4	5	6–10
	POP	**0.739**	0.674	0.690	0.702	0.703	0.710	0.696
	UNN		**0.733**	**0.706**	**0.716**	**0.709**	**0.729**	**0.719**
	INN		0.655	0.674	0.654	0.665	0.672	0.664
	iMF		0.583	0.606	0.630	0.645	0.657	0.624
	HeteRec		0.609	0.623	0.653	0.672	0.680	0.647
Movies	CD-UNN	**0.792**	**0.833**	**0.816**	**0.791**	**0.778**	**0.784**	**0.800**
Movies	CD-INN	0.740	0.676	0.683	0.684	0.680	0.692	0.683
Movies	CD-iMF	0.724	0.660	0.674	0.689	0.686	0.686	0.679
Movies	CD-HeteRec	0.747	0.673	0.672	0.680	0.690	0.704	0.684
Movies	SimMF	0.702	0.649	0.671	0.676	0.682	0.690	0.673
Movies	NeighborMF	0.690	0.652	0.660	0.671	0.680	0.682	0.669
Movies	CentroidMF	0.699	0.647	0.659	0.668	0.684	0.686	0.669
Music	CD-UNN	0.744	**0.811**	**0.797**	**0.771**	**0.746**	**0.734**	**0.772**
Music	CD-INN	**0.746**	0.676	0.683	0.684	0.674	0.689	0.681
Music	CD-iMF	0.720	0.657	0.664	0.674	0.690	0.692	0.675
Music	CD-HeteRec	0.744	0.668	0.655	0.665	0.676	0.687	0.670
Music	SimMF	0.724	0.656	0.675	0.684	0.692	0.692	0.680
Music	NeighborMF	0.721	0.657	0.674	0.684	0.690	0.693	0.679
Music	CentroidMF	0.721	0.655	0.673	0.681	0.692	0.690	0.678

Table 6.10: Diversity (BinomDiv@10) for cold start users in the movies domain.

		Number of movie *likes*						
	Method	0	1	2	3	4	5	6–10
	POP	**0.401**	0.304	0.336	0.354	0.368	0.378	0.348
	UNN		**0.360**	**0.385**	**0.404**	**0.392**	**0.396**	**0.387**
	INN		0.289	0.308	0.315	0.321	0.323	0.311
	iMF		0.299	0.320	0.335	0.344	0.347	0.329
	HeteRec		0.311	0.328	0.334	0.337	0.341	0.330
Books	CD-UNN	**0.467**	**0.509**	**0.479**	**0.446**	**0.425**	**0.414**	**0.455**
Books	CD-INN	0.327	0.291	0.314	0.323	0.329	0.331	0.317
Books	CD-iMF	0.341	0.294	0.317	0.327	0.333	0.338	0.322
Books	CD-HeteRec	0.316	0.310	0.328	0.335	0.337	0.341	0.330
Books	SimMF	0.308	0.265	0.297	0.307	0.320	0.325	0.303
Books	NeighborMF	0.315	0.266	0.298	0.306	0.321	0.325	0.303
Books	CentroidMF	0.313	0.273	0.302	0.315	0.326	0.334	0.310
Music	CD-UNN	**0.368**	**0.404**	**0.386**	**0.376**	**0.373**	**0.372**	**0.382**
Music	CD-INN	0.309	0.240	0.268	0.283	0.297	0.304	0.279
Music	CD-iMF	0.270	0.231	0.270	0.289	0.302	0.315	0.282
Music	CD-HeteRec	0.333	0.271	0.298	0.314	0.324	0.333	0.308
Music	SimMF	0.311	0.254	0.288	0.303	0.317	0.324	0.297
Music	NeighborMF	0.311	0.259	0.290	0.308	0.320	0.329	0.301
Music	CentroidMF	0.302	0.246	0.279	0.297	0.310	0.319	0.290

Table 6.11: Diversity (BinomDiv@10) for cold start users in the music domain.

	Method	0	1	2	3	4	5	6–10
		Number of music *likes*						
	POP	**0.324**	0.228	0.262	0.282	0.295	0.305	0.274
	UNN		**0.296**	**0.332**	**0.348**	**0.347**	**0.330**	**0.331**
	INN		0.200	0.213	0.219	0.223	0.229	0.217
	iMF		0.196	0.217	0.232	0.241	0.249	0.227
	HeteRec		0.227	0.264	0.280	0.288	0.296	0.271
Books	CD-UNN	**0.325**	**0.429**	**0.414**	**0.393**	**0.366**	**0.346**	**0.390**
Books	CD-INN	0.269	0.215	0.227	0.232	0.235	0.240	0.230
Books	CD-iMF	0.270	0.214	0.233	0.240	0.249	0.252	0.237
Books	CD-HeteRec	0.295	0.233	0.271	0.286	0.294	0.302	0.277
Books	SimMF	0.274	0.220	0.240	0.249	0.257	0.264	0.246
Books	NeighborMF	0.254	0.220	0.241	0.251	0.259	0.265	0.247
Books	CentroidMF	0.253	0.218	0.238	0.249	0.257	0.263	0.245
Movies	CD-UNN	0.296	**0.411**	**0.380**	**0.358**	**0.347**	0.329	**0.365**
Movies	CD-INN	0.277	0.231	0.255	0.264	0.270	0.272	0.258
Movies	CD-iMF	0.248	0.229	0.254	0.264	0.271	0.272	0.258
Movies	CD-HeteRec	**0.372**	0.271	0.314	0.331	0.342	**0.349**	0.321
Movies	SimMF	0.225	0.207	0.239	0.250	0.259	0.264	0.244
Movies	NeighborMF	0.252	0.226	0.251	0.265	0.269	0.274	0.257
Movies	CentroidMF	0.264	0.233	0.257	0.270	0.274	0.279	0.263

Diversity of music recommendation

Last, we report the diversity results for music recommendations in Table 6.11. Once again, CD-UNN provides the most diverse recommendations for all music profile sizes in the 1–10 range. However, for completely new users, we highlight the very good performance of CD-HeteRec, which not only is able to generate diverse recommendations, but also achieved the best accuracy results in Table 6.8. The remaining cross-domain approaches are in general worse than single-domain UNN, independently of the exploited source domain. It is also worth noting the contrasting results for CD-INN. While it provides the best performance in terms of accuracy (see Table 6.8), its diversity is the worst for books, and only average for movies.

In summary, we observe a clear trade-off between accurate and diverse recommendations. In general, when approaches perform well in terms of MRR, they tend to suffer in terms of diversity, and vice versa.

6.6 conclusions

Collaborative filtering approaches have become the most investigated and popular solutions to the cross-domain recommendation problem, as they only mine patterns of user-item preferences (i.e., ratings), and do not require any information about the content of the items to bridge the domains of interest. Some other approaches, however, have shown that content-based relations (e.g., based on social tags) can be exploited to bridge the domains more effectively. In this context, recent initiatives such as the Linked Open Data project provide large interconnected repositories of structured knowledge than can be exploited to relate multiple types of data. Such heterogeneous networks allow establishing content-based links between different types of items, and thus providing a new mechanism to bridge domains for cross-domain recommendation.

In this chapter we have exploited Linked Open Data to extract metadata about items in three recommendation domains. Using this additional information, we were able to find relations between items in different domains, and ultimately compute inter-domain item similarities. We then proposed three novel matrix factorization models for cross-domain recommendation that exploit the computed similarities to link knowledge across domains. Experiments in cold start scenarios showed that, depending on the involved source and target domains, cross-domain recommendations exploiting item metadata can be more accurate for users with few preferences in the target domain. However, the improved accuracy comes at the cost of less diversity among the recommendations, and approaches thriving in diversity tend to be less accurate. We argue, nonetheless, that in cold start the priority of a system may be keeping the user engaged by delivering relevant recommendations rather than diverse, non relevant ones.

Regarding the categorization presented in Chapter 3, the models proposed in this chapter belong to the category of **knowledge linkage** cross-domain recommendation approaches. As in previous chapters, we applied our approaches to the **linked-domain exploitation** task with the goal of **addressing the user cold start** problem. In addition to the results reported in this chapter, we conjecture that item metadata may be prove more useful in cross-domain scenarios with low user overlap. In these cases, approaches purely based on collaborative filtering are likely to struggle to compute accurate item-item similarities. Moreover, in our work we relied on advanced Bayesian Optimization techniques to find the optimal hyper-parameters of the models, and in particular the values of the cross-domain regularization λ_C and the item neighborhood size n parameters. It would be interesting, however, to analyze the performance of the models in terms of these parameters to better understand the importance of auxiliary information. We did not report these results in this chapter due to the high number of possible combinations of different parameter values, source-target domain configurations, cold start profile sizes, and cross-validation folds, which may make it very difficult to extract conclusions that consistently hold trough all the possible scenarios.

Part III

DISCUSSION

7 CONCLUSIONS AND FUTURE WORK

In this thesis we have addressed the cold start problem of collaborative filtering in a target domain by means of cross-domain recommendations, generated through the exploitation of different types of auxiliary data from a related source domain. We first proposed a formalization of the cross-domain recommendation problem unifying perspectives in several fields, and a categorization of previous approaches based on their inter-domain knowledge aggregation, linking or transfer processes. Then, we proposed three novel approaches that extend the matrix factorization model for cross-domain collaborative filtering, by exploiting social tags, user personality factors, and item semantically related metadata. Finally, we evaluated the proposed models not only in terms of rating prediction and item ranking accuracy, but also considering alternative dimensions of recommendation quality, namely item novelty and diversity, and domain coverage.

In this chapter we present the main conclusions derived from our research. In Section 7.1 we summarize the work of the thesis and discuss the resulting contributions, and in Section 7.2 we describe potential research issues for future work.

7.1 SUMMARY AND DISCUSSION OF CONTRIBUTIONS

In the subsequent subsections we summarize and discuss the main contributions of this thesis, addressing the research goals stated in Chapter 1. With respect the research goal RG1, we first reviewed the state of the art on cross-domain recommender systems, and provided a unifying formalization of the problem and a categorization of existing approaches. Regarding the research goal RG2, we developed a number of novel matrix factorization models for cross-domain collaborative filtering that exploit different sources of information to bridge recommendation domains, and address the cold start problem in the target domain. Finally, with respect the research goal RG3, we evaluated the proposed models considering multiple aspects of recommendation quality, namely accuracy, novelty, diversity and coverage.

7.1.1 *Formalization of the cross-domain recommendation problem and categorization of existing approaches*

Definition of domain, and cross-domain tasks and goals

In Chapter 3 we presented a novel, exhaustive survey of the state of the art on cross-domain recommender systems. We proposed a formalization of the cross-domain recommendation problem based on a definition of domain that considers several granularity levels, namely item attributes (e.g., movie genres), item types (e.g., movies and TV shows), items (e.g., movies and books), and systems (e.g., MovieLens and Netflix). Our definition generalizes previous notions of domain, and let us unify previous perspectives in which the problem has been addressed. Moreover, we identified the main tasks and goals of the cross-domain recommendation approaches proposed in distinct research fields.

Knowledge aggregation vs. knowledge linkage and transfer

Based on the followed technique to bridge domains, we proposed a categorization of cross-domain recommendation approaches in two main groups. The first group is composed of methods that aggregate knowledge from different domains to perform recommendations in a target domain. From the literature, we identified several types of aggregated knowledge, namely (i) user preferences, (ii) user modeling data such as similarities and neighborhoods, and (iii) recommendation lists. The second group refers to approaches that link or transfer knowledge between domains in order to support the recommendation process. We further classified these approaches into (i) methods that link domains using common knowledge such as semantic networks and inter-domain similarities, (ii) approaches that share latent features learned in each domain, and (iii) approaches that extract and transfer patterns of ratings to the target domain.

Domain overlap scenarios, and evaluation methodologies

In addition to the previous categorization, we also analyzed the required domain overlap setting in each approach, that is, whether it is assumed or not the existence of common users or items in the source and the target domains, to generate recommendations. Finally, we summarized the variety of methodologies that the different works have considered so far for evaluating cross-domain recommendations. We observed that most works focus on the rating prediction task, and measure the performance of the proposed approaches in terms of the accuracy of rating predictions. This led us to investigate in this thesis cross-domain recommendation models for the item ranking task, dealing with positive-only user feedback instead of numerical ratings.

7.1.2 *Cross-domain recommendation models for the cold start*

The main contributions of our research are three novel matrix factorization approaches for cross-domain collaborative filtering, that exploit different sources of auxiliary information to bridge domains and generate cold start recommendations.

Social tag-based matrix factorization

In Chapter 4 we presented our fist matrix factorization model, which exploits social tags shared between the domains of interest to support the transfer of knowledge from the source to the target domain. Our model builds upon previous work (Enrich et al., 2013) that used social tags to link domains and compute rating predictions. By introducing an additional set of latent parameters, the proposed approach separately models the contribution of user and item tags, better capturing their contribution to the observed ratings. The correlation between user/item tags and the ratings is transferred between domains to improve rating predictions for cold start users in the target domain, even if they did not tag the item. We empirically shown that our model achieves the best improvements with respect to previous works, precisely for users with only a few ratings in the target domain.

User personality-based matrix factorization

Instead of exploiting user-item preferences expressed as social tags to bridge the domains, in Chapter 5 we presented a second approach the leverages information about the users' personality factors. Specifically, we proposed matrix factorization models that enhance user profiles with additional latent variables for user attributes based on personality scores from the Five Factor Model, a personality representation well known in Psychology. Our personality-based matrix factorization models were designed to handle positive-only user preferences rather than numerical ratings. For such purpose, we developed an Alternating Least Squares procedure to efficiently train the models. The additional latent variables for personality factors allow transferring knowledge even when there is no overlap between the domains. For the case of user overlap, we further extended our models to exploit user preferences from auxiliary source domains. The results achieved in our experiments shown that the proposed personality-based models are significantly more effective than state of the art approaches for completely new users. However, we observed that once user preferences are available in the target domain, the improvements brought by the exploitation of personality information are only marginal, if any.

Item metadata-based matrix factorization

The previous two approaches respectively leverage social tagging user-item information and user-specific personality information. The models proposed in Chapter 6 utilize item semantic metadata to bridge the domains for cross-domain recommendation with positive-only user feedback. Based on the hypothesis that semantically similar items from different domains should have similar latent parameters, we presented three matrix factorization models that exploit inter-domain semantic similarities to regularize the learning of item factors. The metadata to consider in addition to cross-domain user preferences makes the training of matrix factorization models slow. For this reason, for each of the proposed models, we developed an adaptation of a fast learning algorithm for Alternating Least Squares (Pilászy et al., 2010). The results of our experimental work shown that our models are able to provide more accurate recommendations than graph-based recommendation approaches from the state of the art, depending on the considered recommendation domains and the amount of available user preferences in the target domain.

7.1.3 *Evaluation of recommendation accuracy, novelty and diversity, and domain coverage*

As previously mentioned, our analysis of the state of the art on cross-domain recommender systems showed that most of the existing approaches have been evaluated by measuring the error of rating predictions. In this thesis we also considered other dimensions of recommendation quality to compare the performance of the proposed models.

Evaluating rating predictions of the social tag-based model

Aiming to fairly compare our tag-based matrix factorization model, in Chapter 4 we attempted to reproduce the evaluation setting and methodology followed by the previous work upon which our model is based (Enrich et al., 2013). Therefore, we evaluated the proposed model in the same task of rating prediction, on the same dataset, and using the same error-based metric. The evaluated models were able to transfer knowledge between the domains even when there is no user or item overlap, since only tag overlap is required. Hence, the datasets considered do not share users or items. In order to simulate increasing levels of cold start, we followed the methodology proposed in (Enrich et al., 2013) to randomly downsample different amounts of observed data in the target domain, which in turn results in several levels of tag overlap. Our results shown that our model is most effective in severe cold start situations with lower tag overlap. Moreover, we also analyzed the accuracy of the models for users with different number of observed target ratings and tags, and observed that our model achieves the best improvements with respect to the state of the art for users with few ratings and tags.

Evaluating recommendation ranking of the personality- and metadata-based models

In Chapter 5 and Chapter 6 we evaluated our personality- and metadata-based models with different datasets composed of positive-only feedback in the form of Facebook *likes*, as opposed to numerical ratings. As a result, we measured the performance of our models in the item ranking or top-N recommendation task, using ranking accuracy metrics such as MAP and MRR, rather than error-based metrics such as MAE and RMSE as previously done in the literature. Moreover, we followed a principled evaluation methodology designed for cold start situations (Kluver and Konstan, 2014), and extended it to deal with the cross-domain setting, and to generate validation sets that we used for tuning the model parameters and evaluating baseline preference elicitation strategies.

Moreover, in Chapter 5 we evaluated the proposed personality-based matrix factorization models in two settings with and without user overlap. We also analyzed the item catalog coverage or spectrum of recommended items, and the entropy of the item distribution as a measure of aggregate diversity. The achieved empirical results for the personality-based model shown that it is able to provide relevant recommendations to completely new users, and its suggested items are significantly different than the most popular recommendations. Moreover, in Chapter 6 we went one step further, and extracted item metadata, which allowed us to compute similarities between the recommended items, and eventually to measure the diversity of the recommendation lists. In this context, we observed that there was often a trade-off between accuracy and diversity in cold start settings, and that approaches that perform well in terms of accuracy, in some situations struggle to provide diverse recommendations. Conversely, in the same situations, approaches that provide very good diversity usually do so by recommending non relevant items.

7.2 OPEN RESEARCH ISSUES

Adequacy of auxiliary data sources

As we have shown in this thesis, auxiliary data from related source domains can be exploited to deliver relevant recommendations for cold start users in the target domain. From our experiments, however, we have also shown that the additional information may not be helpful, and even harmful, when enough user preferences are available in the target domain. Throughout the thesis, we have observed this phenomenon *a posteriori* by analyzing the performance of recommendation approaches for different cold start profile sizes. In real applications, in contrast, a system should decide in advance whether the source domain preferences are worth being exploited or not. An intelligent method that automatically decides if the source domain preferences should be considered is therefore a desirable component of cross-domain recommender systems.

In Machine Learning, three different aspects have been proposed to be considered for the development of transfer learning models (Pan and Yang, 2010):

- *What to transfer?* The first aspect is related to the specific knowledge that is transferred between the models. In this thesis we have proposed three matrix factorization models that respectively transfer latent factors for social tags, user personality, and item semantic metadata.
- *How to transfer?* The second aspect refers to the machine learning approach that is used to make the transfer of knowledge. The methods proposed in Chapter 4 and Chapter 5 share the latent vectors of social tags and personality factors, respectively, and in Chapter 6 the transfer is conducted through the regularization of item factors with item similarity information during the learning process.
- *When to transfer?* The third aspect deals with the problem of *negative transfer*, i.e., the situation in which the transferred knowledge is not helpful for the task at hand. A key component in

the success of these approaches is deciding when the auxiliary information should be transferred.

The *when to transfer* issue directly connects with our previous observation regarding the availability of target domain preferences. Therefore, as a possible improvement for the matrix factorization models proposed in this thesis, we envision the application of machine learning techniques to automatically avoid the problem of *negative transfer*. In particular, probabilistic models using Bayesian inference (Salakhutdinov and Mnih, 2008) provide a good way to address the problem, as they are able to compute confidence estimations on the learned parameters. Few user preferences in the target domain will result in poorly estimated user factors with very low confidence, hinting that source domain knowledge should be exploited. As more target preferences become available, the uncertainty will gradually decrease up to the point that user factors learned only with target data are accurate enough for recommendation.

Processing of social tags

Regarding the social tag-based recommendation models discussed in Chapter 4, previous work by Enrich et al., (2013) showed that discarding non relevant tags may lead to better accuracy. Our TagGSVD++ model, in contrast, is able to outperform the baselines without taking tag relevance into account by separately modeling the effect of user and item tags on the ratings. Nonetheless, it is likely that even better results could be achieved by carefully selecting the most promising tags for recommendation. Moreover, we did not perform any sophisticated preprocessing of the tags in our dataset. According to the works of Szomszor et al., (2008a) and Abel et al., (2011), we conjecture that the alignment of social tags from different domains based on their semantics can be useful to increase the overlap between the domains and ultimately the quality of the recommendations.

Representation of user personality

In Chapter 5 we enhanced matrix factorization with attributes that represent personality scores from the Five Factor Model, a representation popular in Psychology that considers five orthogonal factors of personality. We tested several discretization strategies to transform the scores into variables for the proposed recommendation models, and the achieved empirical results showed that personality information is only useful for completely new users with no target preferences at all. However, it may be the case that alternative models of personality better correlate with users preferences, and thus may be more suitable for recommendation. For instance, better results could be obtained by exploiting more fine-grained representations based on personality facets, e.g., the *imagination*, *artistic interests*, and *emotionality* facets of the *openness* factor are likely to be of importance when discovering relationships between user preferences and personality. Moreover, through the chapter we assumed that personality information was available. In real applications, such data have to be acquired, usually by asking the users to fill questionnaires, based on which their FFM scores are estimated. This is a notable limitation in cold start scenarios, where users that just registered in the system expect to receive recommendations, and may not be willing to spend time providing much information. In this context, a possible alternative is to obtain personality information implicitly by analyzing the users' behavioral patterns during the interaction with the system or others (Kosinski et al., 2013).

Selection of semantic features

With respect to Chapter 6, the proposed cross-domain matrix factorization models that exploit item semantic metadata are based on inter-domain semantic similarities that are computed using the link structure of semantic networks in Wikipedia/DBpedia. Such similarities, however, do not take into account the different relevance of the semantic properties that relate the domains of interest. Recent work (Musto et al., 2016b) has shown that a careful selection of relevant features may lead to better performance while reducing the size of the semantic networks, which results in more efficiency for the recommendation algorithms.

User studies

Finally, we note that all the experimental work in this thesis was conducted in *offline* settings, using datasets that contain user preferences in several domains. To the best of our knowledge, there is no publicly available dataset that contains user preferences and the three sources of data considered in the thesis, namely social tags, personality factors, and semantic annotations. It would be interesting to determine which of the considered data sources is more valuable for cross-domain recommendation in cold start scenarios, but the lack of such a dataset makes impossible to compare the models proposed in this thesis against each other. A potential direction for future work is to perform a user study collecting such information from a large enough group of users, and requesting their assessment of recommendations generated with each model.

Part IV

APPENDICES

A

INTRODUCCIÓN

A.1 MOTIVACIÓN

Sobrecarga de información en la Web

En la última década se ha producido un aumento exponencial del número de recursos disponibles en la *World Wide Web*, sobre todo desde que ésta se hizo ampliamente accesible y los usuarios comenzaron a crear y cargar en ella sus propios contenidos. Llegando a millones de usuarios, sitios de comercio electrónico como Amazon.com y eBay venden cientos de millones de productos pertenecientes a docenas de categorías, proveedores de contenidos multimedia por internet como Spotify y Netflix ofrecen acceso a enormes catálogos de música y programas de televisión, y se estima que los usuarios de YouTube producen más de 400 horas de vídeo cada minuto. Es más, las redes sociales en línea representan la mayoría de las actividades de los usuarios en la Web, como Facebook con más de 300 millones de fotos compartidas cada día.

Sistemas de recomendación

En este contexto, la ingente y continuamente creciente cantidad de contenidos conlleva un problema de *sobrecarga de información*, ya que encontrar ítems de información relevantes en grandes colecciones puede resultar una tarea demasiado compleja y costosa para las personas. Los sistemas de recomendación son herramientas software diseñadas para ayudar a los usuarios en sus tareas de acceso y recuperación de información. Analizando interacciones previas de los usuarios con ciertos ítems, estos sistemas infieren las preferencias de los usuarios por otros ítems para predecir y sugerir los más relevantes. Son de este modo componentes esenciales de muchos servicios de negocio, educación, cultura y entretenimiento.

Arranque en frío

En el mundo académico, los sistemas de recomendación se han estudiado activamente desde los años noventa y actualmente representan un área de investigación consolidada, como lo demuestra la ACM Conference on Recommender Systems[1], que después de 10 ediciones se ha convertido en un foro internacional de muy alto prestigio. En los últimos años, múltiples enfoques de recomendaciones, y notablemente los basados en el filtrado colaborativo, se han propuesto e implementado exitosamente. Sin embargo, todavía existen desafíos y limitaciones que ofrecen oportunidades de investigación. Una de las más notorias de estas oportunidades es la del problema del arranque en frío (*cold start* en inglés), que se refiere a la situación en la que un nuevo usuario se ha registrado recientemente en un sistema, y para el que no hay suficientes preferencias con las que proporcio-

1 ACM Conference on Recommender Systems, RecSys, `http://recsys.acm.org`

nar recomendaciones personalizadas relevantes. El arranque en frío ha recibido mucha atención en la comunidad científica e industria, ya que proporcionar buenas recomendaciones a nuevos usuarios es fundamental para mantenerlos comprometidos con el sistema; si los elementos sugeridos no son relevantes, los usuarios pueden percibir el sistema como no útil y abandonarlo.

Dos principales tipos de soluciones se han explorado para abordar el problema del arranque en frío. El primero está representado por técnicas que tienen como objetivo la adquisición inteligente de preferencias de usuario, pidiendo directamente a los usuarios que evalúen una limitada selección de ítems de información. El segundo incluye métodos que hacen uso de datos auxiliares para inferir preferencias de usuario. Dentro de este último tipo de enfoques, la recomendación sobre dominios cruzados (*cross-domain recommendation* en inglés) ha emergido recientemente como una potencial solución, explotando preferencias de usuario y atributos de ítem en dominios diferentes, pero relacionados con el dominio de recomendación de destino.

Recomendaciones sobre dominios cruzados

La recomendación sobre dominios cruzados es un tema de investigación emergente en varias áreas, con objetivos y tareas particulares. Trabajos sobre Modelado de Usuario han propuesto enfoques sobre dominios cruzados como un mecanismo para agregar y mediar perfiles de usuario de diferentes dominios en estrategias de personalización entre sistemas. En el campo de Aprendizaje Automático, el filtrado colaborativo sobre dominios cruzados se ha investigado como una aplicación práctica de técnicas de aprendizaje por transferencia (*transfer learning* en inglés), que pretenden usar modelos aprendidos a partir de conjuntos de datos con diferentes características y distribuciones. Por último, en Sistemas de Recomendación, los enfoques de dominios cruzados se han estudiado principalmente como un mecanismo para mitigar la escasez de preferencias de usuario en un dominio de destino.

Exhaustiva revisión del estado del arte en sistemas de recomendación sobre dominios cruzados

Esta diversidad de objetivos, tareas y enfoques ha dado lugar a múltiples formulaciones complementarias del problema de recomendación sobre dominios cruzados. Además, ha provocado que no exista un consenso en la definición de dominio de recomendación, dificulta la clasificación y comparación de las soluciones propuestas en la literatura, y la identificación de nuevas oportunidades de investigación. Una primera contribución de esta tesis es la revisión exhaustiva del estado del arte en recomendación sobre dominios cruzados en las áreas citadas anteriormente, proporcionando una formalización integradora del problema, así como una categorización de los métodos de recomendación y metodologías de evaluación.

Novedosos modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados

Un importante aspecto en el desarrollo de sistemas de recomendación sobre dominios cruzados es la manera en la que se establece un *puente* que permita la agregación o transferencia de conocimiento desde un dominio fuente auxiliar hasta un dominio de destino. La mayoría de los enfoques propuestos hasta la fecha se centran en soluciones de filtrado colaborativo que sólo explotan las preferencias de usuario en forma de valoraciones numéricas (ratings). Esto tiene la ventaja de que no se requiera otro tipo de datos adicionales sobre los usuarios o los ítems, que pueden ser muy heterogéneos entre dominios. Sin embargo, puede ser una limitación si no hay usuarios o ítems comunes a los dominios. Además, como ya se ha mostrado en varios estudios, la información auxiliar sobre los usuarios o el contenido de los ítems puede conducir a recomendaciones más efectivas.

En esta dirección, esta tesis presenta nuevas extensiones del método de factorización matricial para filtrado colaborativo (Hu et al., 2008; Koren, 2008). En concreto, propone modelos de factorización matricial para el filtrado colaborativo sobre dominios cruzado que tienen como objetivo mitigar el problema del arranque en frío, mediante la explotación de tres fuentes de información diferentes, a saber, etiquetas sociales, factores de personalidad de los usuarios, y metadatos semánticos de los ítems.

Etiquetas sociales

Durante los últimos años, ha habido una creciente popularización de los servicios de etiquetado social, en los que los usuarios crean contenidos y los anotan con palabras libremente elegidas conocidas como etiquetas (*tags* en inglés). El conjunto de etiquetas de cada sistema constituye un esquema de clasificación de conocimiento colaborativo y no estructurado, que puede considerarse como una fuente de preferencias de usuario, ya que los usuarios asignan etiquetas a contenidos propios y a contenidos de otros que les son de interés, y que por tanto puede usarse con fines de recomendación.

Factores de personalidad

Alternativamente, la personalidad es un patrón de valores, actitudes y conductas que caracterizan a las personas, y tiene cierta persistencia a lo largo de la vida, de modo que las manifestaciones de ese patrón en diferentes situaciones tienen cierto grado de previsibilidad. Así, en algunos dominios, se ha mostrado que las personas con rasgos de personalidad similares tienden a tener preferencias parecidas, lo que hace que la personalidad sea una potencial fuente de información para proporcionar recomendaciones de filtrado colaborativo.

Metadatos semánticos

Además del filtrado colaborativo, el filtrado basado en contenido se ha aplicado en dominios donde el contenido y metadatos de ítems desempeñan un papel clave, tanto de manera adicional como alternativa a valoraciones explícitas y retroalimentación implícita de los usuarios. Con el advenimiento de la Web Semántica, y su implementación de referencia *Linked Data*, una gran cantidad de metadatos estructurados y enlazados está disponible en la Web. Estos metadatos también representan así una potencial fuente de información a ser explotada por aproximaciones de filtrado basado en contenido y recomendaciones híbridas.

Retroalimentación sólo positiva como preferencias de usuario

Por otra parte, a diferencia de la mayoría de los trabajos previos en el estado del arte, que se han centrado en la tarea de predicción de ratings, dos de los tres enfoques propuestos en esta tesis están diseñados para utilizar retroalimentación positiva como fuente de preferencias de usuario en la tarea de ranking de ítems. éste es posiblemente un escenario más realista, ya que las retroalimentaciones positivas (e.g., logs de clics, contadores de consumo, y registros de compra) se recogen fácil e implícitamente por el sistema. Sin embargo, a menudo son más difíciles de explotar, ya que la retroalimentación obtenida sólo sirve como evidencia de gustos de los usuarios, pero no proporciona ninguna información sobre sus aversiones.

Resultados en tres dominios con conjuntos de datos grandes

En el trabajo experimental de la tesis, se comparan empíricamente los modelos propuestos usando conjuntos de datos grandes que abarcan varios dominios, a saber, recomendaciones de películas, música y libros. Los resultados obtenidos muestran que los modelos propuestos son efectivos en escenarios de arranque en frío, no sólo en términos de precisión de las estimaciones de relevancia de los ítems, sino también con respecto a la novedad y diversidad de las recomendaciones y la cobertura de los dominios.

A.2 OBJETIVOS

Objetivos de investigación de la tesis

En esta tesis se propone investigar cómo las recomendaciones sobre dominios cruzados pueden usarse para mitigar el problema del arranque en frío en filtrado colaborativo. Para ello se plantea la hipótesis de que la explotación de información auxiliar, adicional a las preferencias de usuario, sobre usuarios e ítems permite una transferencia de conocimiento entre dominios más efectiva y, por lo tanto, la generación de mejores recomendaciones en situaciones de escasez de preferencias de usuario. Con esta hipótesis se establecen los siguientes objetivos de investigación específicos.

O1: revisar el estado del arte en sistemas de recomendación sobre dominios cruzados, con el fin de identificar trabajos relacionados que aborden el arranque en frío. Como se mencionó en la sección anterior, la recomendación sobre dominios cruzados se ha tratado en varias áreas, y aún no existe consenso en cuanto a la formalización del problema y una visión global de los objetivos y tareas para las cuales las diferentes soluciones han sido diseñadas. El primer objetivo de la tesis es realizar un estudio riguroso y exhaustivo de la literatura para unificar perspectivas e identificar oportunidades de investigación.

O2: desarrollar nuevos modelos de recomendación sobre dominios cruzados que exploten la información auxiliar además de las preferencias del usuario, y evaluarlos rigurosamente en situaciones de arranque en frío. La mayoría de los enfoques sobre dominios cruzados propuestos hasta la fecha se basan en métodos de filtrado colaborativo que sólo consideran las preferencias de usuario. En esta tesis, en cambio, se plantea utilizar otros tipos de información sobre usuarios e ítems. La revisión realizada en O1 permitiría determinar las posibles fuentes de datos que podrían beneficiar al filtrado colaborativo sobre dominios cruzados en situaciones de arranque en frío. Para validar las soluciones desarrolladas, se requiere que la evaluación de los modelos se realice en varios dominios y con conjuntos de datos relativamente grandes. También se establece el seguir una metodología de evaluación adecuada para el arranque en frío. De nuevo, la revisión de O1 ayudaría en estas cuestiones.

O3: analizar la eficacia de los modelos de recomendación sobre dominios cruzados yendo más allá de su precisión. En la literatura, la mayoría de los enfoques de recomendación sobre dominios cruzados existentes se han evaluado en términos del error en las predicciones de ratigs explícitos. De acuerdo con la naturaleza de los modelos de recomendación desarrollados en O2, en esta tesis se plantea abordar tanto la predicción de ratings como la generación de ranking de ítems, y considerar diversos tipos de preferencias de usuario. Por estas razones, el tercer objetivo de la tesis es evaluar los modelos mediante el acierto de predicciones de relevancia de ítem y la precisión de listas de recomendaciones, apropiadas para la retroalimentación numérica, binaria o positiva. Además, se plantea analizar la efectividad de los modelos propuestos según las propiedades de recomendación distintas de la precisión, como la novedad, la diversidad y la cobertura. Análogamente a O2, la revisión de la literatura de O1 permitiría elegir las métricas de evaluación adecuadas.

A.3 CONTRIBUCIONES

Principales contribuciones de la tesis

El trabajo realizado en esta tesis ha dado lugar a varias contribuciones al estado del arte en sistemas de recomendación sobre dominios cruzados, que se resumen a continuación.

En el Capítulo 3 se ofrece un análisis exhaustivo y en profundidad de trabajos previos en sistemas de recomendación sobre dominios cruzados. En la tesis se presenta **una formalización del problema** considerando una definición integradora de dominio de recomendación en diferentes niveles de granularidad, para **unificar nociones de dominio** usadas en la literatura. Además, se identifican las diferentes tareas abordadas en el estado del arte, así como los objetivos de recomendación perseguidos, e.g., mejorar la precisión de las predicciones de relevancia de ítem, enriquecer los modelos de usuario, y mitigar el arranque en frío. Finalmente, **categoriza los enfoques de recomendación existentes**, distinguiendo modelos que agregan las preferencias de usuario y modelos que vinculan o transfieren conocimiento del dominio de origen al dominio de destino.

En el Capítulo 4 se presenta una **extensión del método de factorización matricial que incorpora parámetros adicionales para modelar y transferir el efecto de etiquetas sociales en ratings entre dominios**. Se revisan enfoques anteriores basados en etiquetado social para la recomendación sobre dominios cruzados con el fin de identificar sus fortalezas y limitaciones. Se toma como punto de partida un modelo de factorización matricial para dominios individuales que explota metadatos de ítem y se adapta al escenario de varios dominios. El enfoque propuesto en la tesis modela por separado la contribución de las etiquetas de los usuarios y de los ítems, permitiendo captar mejor su efecto sobre los ratings observados y calcular predicciones de relevancia incluso cuando el usuario no ha etiquetado el ítem destino.

En el Capítulo 5 se presentan **modelos de factorización matricial basados en personalidad** que explotan información sobre factores de personalidad de los usuarios para calcular recomendaciones para nuevos usuarios en configuraciones de dominio único y de dominios cruzados. Los modelos planteados están diseñados para **manejar retroalimentación sólo positiva**, en vez de ratings numéricos. Además, para ellos se estudian varios **métodos de modelado de la personalidad de los usuarios**, y se implementa una **extensión del algoritmo de mínimos cuadrados alternados** para un entrenamiento eficiente con la información adicional considerada.

Por último, en el Capítulo 6 se presentan tres **extensiones de la factorización matricial para filtrado colaborativo sobre dominios cruzados que explotan metadatos de ítem semánticamente relacionados, como puente de enlace entre dominios**. En particular, dicha información semántica se utiliza para calcular las similitudes inter-dominio para los ítems, que posteriormente se utilizan para regularizar los factores latentes. Para los tres modelos propuestos, se proporcionan **algoritmos de entrenamiento eficientes para aprender los parámetros óptimos**, basados en una versión rápida de mínimos cuadrados alternados propuesta en la literatura (Pilászy et al., 2010).

A.4 ESTRUCTURA DEL DOCUMENTO

Tres capítulos presentando las soluciones propuestas en la tesis

En esta tesis se propone una serie de nuevos modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados, que resultan efectivos en mitigar el arranque en frío en un dominio de destino, mediante la explotación de datos de un dominio fuente auxiliar distintos a las preferencias de usuario proporcionadas como ratings numéricos. En particular, se investiga la explotación de interacciones de los usuarios con ítems en forma de etiquetas sociales, factores de personalidad de los usuarios, y metadatos de ítem semánticamente relacionados. Estas soluciones se exponen en tres capítulos centrales de este documento.

La variedad anterior de fuentes de información implica tratar temas de investigación de varias áreas, tales como la minería de etiquetado social en Inteligencia Artificial, el modelado de personalidad en Psicología y la representación de conocimiento en la Web Semántica. Además, estos tipos de información también han originado la propuesta de enfoques particulares en el ámbito de los Sistemas de Recomendación, a saber, métodos de recomendación basados en etiquetado social, personalidad y semántica, tanto en situaciones de dominio ínico como de dominios cruzados. Teniendo en cuenta que una descripción muy extensa del estado del arte en todos estos temas y áreas puede resultar abrumadora para el lector, se han distribuido reseñas bibliográficas específicas en los tres capítulos correspondientes. Estos tres capítulos tienen la misma estructura, con secciones para introducir y motivar la investigación realizada, examinar enfoques existentes, presentar los modelos de recomendación propuestos, y reportar y analizar los resultados obtenidos en los experimentos realizados.

Dos capítulos analizando el contexto y antecedentes de la tesis

A pesar de que cada uno de estos capítulos centrales trata trabajos relacionados particulares, con el fin de ofrecer una visión global del contexto y antecedentes de la tesis, en una primera parte del documento, se han dedicado dos capítulos a una descripción de aspectos generales de los sistemas de recomendación, y una revisión exhaustiva de los sistemas de recomendación sobre dominios cruzados.

Contenido específico de cada capítulo

El contenido de todos los capítulos se detalla a continuación.

Parte I: Contexto y antecedentes

- El Capítulo 2 proporciona una visión general de diversos aspectos de los **sistemas de recomendación**. En el capítulo primero se plantea el problema de recomendación, distinguiendo las tareas de predicción de relevancia de ítems y de ranking de ítems, se explican los principales tipos de preferencias de usuario, y se discute el problema de la escasez de preferencias de usuario en situaciones de arranque en frío. A continuación, se da una categorización y descripción de técnicas de recomendación generales, a saber, filtrado basado en contenido y filtrado colaborativo, y se detalla la factorización matricial para filtrado colaborativo, que es la base de los modelos de recomendación propuestos en la tesis. Finalmente, se describen metodologías y métricas para evaluar sistemas de recomendación, algunos de los cuales han sido utilizados en el trabajo experimental de la tesis.

- El Capítulo 3 presenta una novedosa y exhaustiva revisión del estado del arte en **sistemas de recomendación sobre dominios cruzados**. Unificando perspectivas de diferentes áreas, primero se propone una formulación del problema, tareas y objetivos. Posteriormente, se propone una categorización de las técnicas de recomendación, distinguiendo métodos de agregación de conocimiento y métodos de enlace y transferencia de conocimiento. Para cada uno de estos tipos de técnicas, se analiza y compara un gran número de enfoques existentes. De manera análoga al capítulo anterior, se concluye con una discusión sobre cuestiones relativas a la evaluación de sistemas de recomendación sobre dominios cruzados.

Parte II: Soluciones propuestas

- El Capítulo 4 propone un modelo de factorización matricial para filtrado colaborativo sobre dominios cruzados que explota **etiquetas sociales** como fuente de preferencias de usuario compartidas o relacionadas entre diferentes dominios. En el capítulo se revisan enfoques de recomendación basados en etiquetado social para dominios únicos y cruzados, centrándose en modelos de factorización matricial que han inspirado el propuesto en el capítulo. A continuación, se describe el modelo propuesto, y se reportan y analizan resultados empíricos obtenidos a partir de la evaluación del modelo para la tarea de predicción de ratings en situaciones de arranque en frío, utilizando los bien conocidos conjuntos de datos MovieLens[2] y LibraryThing[3], para los dominios de recomendación de películas y libros.

- El Capítulo 5 propone modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados que consideran **factores de personalidad de los usuarios** como características independientes de dominio, y los explotan para establecer relaciones entre preferencias de usuario sobre ítems de dominios diferentes. En el capítulo primero se motiva el enfoque propuesto revisando trabajos previos que han mostrado la existencia de relaciones entre factores de personalidad y preferencias de usuario en ciertos dominios, y analizando enfoques existentes que han incorporado información de personalidad en heurísticas de filtrado colaborativo. Posteriormente, se presenta la propuesta de modelo de factorización matricial basado en personalidad que, de manera diferente a enfoques existentes, es evaluado con un conjunto de datos grande en tres dominios, a saber, recomendaciones de películas, música y libros. Más específicamente, se reportan y analizan resultados empíricos obtenidos con un conjunto de datos extraído del proyecto myPersonality[4], que proporciona un gran número de perfiles de usuario compuestos por *likes* de Facebook[5] y valores de los factores *Big Five* de personalidad. El modelo se evalúa de este modo con datos de retroalimentación sólo positiva, para la tarea de ranking de ítems.

2 Conjuntos de datos de MovieLens, `http://grouplens.org/datasets/movielens`
3 Conjunto de datos de LibraryThing, `http://www.macle.nl/tud/LT`
4 Proyecto myPersonality, `http://mypersonality.org`
5 Red social Facebook, `https://www.facebook.com`

- El Capítulo 6 propone modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados que, en lugar de explotar datos de usuario en forma de etiquetas sociales y datos específicos sobre la personalidad de los usuarios, se centran en el uso de **metadatos semánticos de los ítems** para vincular las preferencias de los usuarios por ítems de dominios diferentes. En particular, los modelos propuestos hacen uso de atributos y relaciones semánticas extraídas automáticamente de DBpedia[6], que es la versión estructurada de la popular Wikipedia[7], y la principal base de conocimiento del proyecto Linked Open Data[8]. En el capítulo se examinan los enfoques de recomendación del estado del arte que explotan Linked Data y se presentan los modelos de factorización matricial propuestos. De forma similar a los modelos basados en personalidad, para evaluar las propuestas basadas en semántica, se realizan experimentos para la tarea de ranking de ítems en el arranque en frío, con un conjunto de datos compuesto únicamente por *likes* de Facebook. En este caso, los ítems se enlazan automáticamente a entidades DBpedia, cuyos metadatos se extraen y se utilizan para crear redes semánticas que enlazan los ítems entre dominios.

- El Capítulo 7 finaliza la tesis con **conclusiones** generales sobre la explotación de información adicional a las preferencia de usuario mediante modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados. En el capítulo también se discuten las limitaciones y problemas pendientes de investigación no abordados en la tesis, que puede motivar investigaciones futuras.

A.5 PUBLICACIONES

El trabajo presentado en esta tesis ha resultado en varias publicaciones de revista, un libro, y congresos y talleres internacionales. A continuación se listan estos artículos científicos, agrupados y ordenados de acuerdo con los capítulos de este documento y los temas de investigación de la tesis con los que están relacionados.

Publicaciones relacionadas con el Capítulo 3, Sistemas de recomendación sobre dominios cruzados

Publicaciones presentando revisiones de sistemas de recomendación sobre dominios cruzados

Las primeras contribuciones de esta tesis son la formalización del problema de recomendación sobre dominios cruzados –unificando las perspectivas desde las que se ha tratado– y la categorización analítica, descripción y comparación de trabajos previos –realizando una revisión exhaustiva de un gran número de artículos en diferentes

6 Repositorio de conocimiento DBpedia, http://wiki.dbpedia.org
7 Enciclopedia en línea Wikipedia, https://www.wikipedia.org
8 Proyecto Linked Open Data, http://linkeddata.org

áreas de investigación, a saber, Modelado de Usuario, Aprendizaje Automático y Sistemas de Recomendación. Las siguientes publicaciones presentan tales contribuciones:

- Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, Paolo Cremonesi. 2015. **Cross-domain Recommender Systems**. En *Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.), Recommender Systems Handbook - 2nd edition*, pp. 919-959. Springer, ISBN 978-1-4899-7636-9.

- Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskas, Francesco Ricci. 2012. **Cross-domain Recommender Systems: A Survey of the State of the Art**. En *Actas de 2nd Spanish Conference on Information Retrieval (CERI 2012)*, pp. 187-198. Publicaciones de la Universitat Jaume I, ISBN 978-84-8021-860-32.

Publicaciones relacionadas con el Capítulo 4, Modelos de factorización matricial para filtrado colaborativo basados en etiquetas sociales

Publicaciones acerca de recomendación sobre dominios cruzados basada en etiquetdo social

En la tesis se valida empíricamente la hipótesis de que las etiquetas sociales pueden ser utilizadas para establecer relaciones entre las preferencias y sentimientos de los usuarios acerca de ítems de dominios diferentes, y que tales relaciones pueden ser explotadas para proporcionar recomendaciones. La propuesta y evaluación de enfoques de modelado de usuario y factorización matricial para recomendación sobre dominios cruzados usando etiquetas sociales se presentan en las siguientes publicaciones:

- Ignacio Fernández-Tobías, Iván Cantador. 2014. **Exploiting Social Tags in Matrix Factorization Models for Cross-domain Collaborative Filtering**. En *Actas de 1st International Workshop on New Trends in Content-based Recommender Systems (CBRecSys 2014)*, pp. 34-41. CEUR Workshop Proceedings 1245, ISSN 1613-0073.

- Ignacio Fernández-Tobías, Iván Cantador, Laura Plaza. 2013. **A Social Tag-based Dimensional Model of Emotions: Building Cross-domain Folksonomies**. *Procesamiento del Lenguaje Natural* 51, pp. 195-202. Sociedad Española de Procesamiento del Lenguaje Natural, ISSN 1135-5948.

- Ignacio Fernández-Tobías, Iván Cantador, Laura Plaza. 2013. **An Emotion Dimensional Model Based on Social Tags: Crossing Folksonomies and Enhancing Recommendations**. En *Actas de 14th International Conference on Electronic Commerce and Web Technologies (EC-WEB 2013)*, pp. 88-100. Lecture Notes in Business Information Processing 152, Springer, ISBN 978-3-642-39877-3.

Publicaciones relacionadas con el Capítulo 5, Modelos de factorización matricial para filtrado colaborativo basados en factores de personalidad

Publicaciones acerca de recomendación sobre dominios cruzados basada en personalidad

En la tesis se explotan las relaciones existentes entre factores de personalidad y preferencias de usuario para ítems pertenecientes a diferentes dominios, proponiendo métodos heurísticos y modelos de factorización matricial basados en la personalidad de los usuarios para filtrado colaborativo en dominios únicos y cruzados. Estos enfoques de recomendación, junto con un estudio previo sobre las relaciones anteriores, se recogen en las siguientes publicaciones:

- Ignacio Fernández-Tobías, Matthias Braunhofer, Mehdi Elahi, Francesco Ricci, Iván Cantador. 2016. **Alleviating the New User Problem in Collaborative Filtering by Exploiting Personality Information**. *User Modeling and User-adapted Interaction* 26(2), pp. 221-255. Springer, ISSN 0924-1868.

- Ignacio Fernández-Tobías, Iván Cantador. 2015. **On the Use of Cross-Domain User Preferences and Personality Traits in Collaborative Filtering**. En *Actas de 23rd International Conference on User Modeling, Adaptation, and Personalization (UMAP 2015)*, pp. 343-349. Lecture Notes in Computer Science 9146, Springer, ISBN 978-3-319-20266-2.

- Ignacio Fernández-Tobías, Iván Cantador. 2014. **Personality-aware Collaborative Filtering: An Empirical Study in Multiple Domains with Facebook Data**. En *Actas de 15th International Conference on Electronic Commerce and Web Technologies (EC-Web 2014)*, pp. 125-137. Lecture Notes in Business Information Processing 188, Springer, ISBN 978-3-319-10490-4.

- Iván Cantador, Ignacio Fernández-Tobías. 2014. **On the Exploitation of User Personality in Recommender Systems**. En *Actas de 1st International Workshop on Decision Making and Recommender Systems (DMRS 2014)*. CEUR Workshop Proceedings 1278, ISSN 1613-0073.

- Iván Cantador, Ignacio Fernández-Tobías, Alejandro Bellogín. 2013. **Relating Personality Types with User Preferences in Multiple Entertainment Domains**. En *Late-Breaking Results, Project Papers and Workshop Proceedings of the 21st Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*. CEUR Workshop Proceedings 997, ISSN 1613-0073.

Publicaciones relacionadas con el Capítulo 6, Modelos de factorización matricial para filtrado colaborativo basados en metadatos de ítem

Publicaciones acerca de recomendación sobre dominios cruzados basada en metadatos de ítem

En la tesis se investiga la explotación de metadatos de ítem para establecer relaciones entre dominios, y la incorporación de tales relaciones en modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados. El proceso de extracción de metadatos de ítem de repositorios Linked Data, la construcción de redes semánticas que enlazan ítems en múltiples dominios, y la propuesta y evaluación de los modelos de recomendación propuestos se presentan en los siguientes publicaciones:

- Ignacio Fernández-Tobías, Paolo Tomeo, Iván Cantador, Tommaso Di Noia, Eugenio Di Sciascio. 2016. **Accuracy and Diversity in Cross-domain Recommendations for Cold-start Users with Positive-only Feedback**. En *Actas de 10th ACM Conference on Recommender Systems (RecSys 2016)*, pp. 119-122. ACM, ISBN 978-1-4503-4035-9.

- Ignacio Fernández-Tobías, Roi Blanco. 2016. **Memory-based Recommendations of Entities for Web Search Users**. En *Actas de 25th ACM International Conference on Information and Knowledge Management (CIKM 2016)*, pp. 35-44. ACM, ISBN 978-1-4503-4073-1.

- Paolo Tomeo, Ignacio Fernández-Tobías, Tommaso Di Noia, Iván Cantador. 2016. **Exploiting Linked Open Data in Cold-start Recommendations with Positive-only Feedback**. En *Actas de 4th Spanish Conference on Information Retrieval (CERI 2016)*, art. 11. ACM, ISBN 978-1-4503-4141-7.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2014. **Knowledge-based Identification of Music Suited for Places of Interest**. *Journal of Information Technology and Tourism* 14(1), pp. 73-95. Springer, ISSN 1098-3058.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2013. **Ontology-based Identification of Music for Places**. En *Actas de 13th International Conference on Information and Communication Technologies in Tourism (ENTER 2012)*, pp. 436-447. Springer, ISBN 978-3-642-36308-5.

- Marius Kaminskas, Ignacio Fernández-Tobías, Francesco Ricci, Iván Cantador. 2012. **Knowledge-based Music Retrieval for Places of Interest**. En *Actas de 2nd International Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies (MIRUM 2012)*, pp. 19-24. ACM, ISBN 978-1-4503-1591-3.

- Ignacio Fernández-Tobías, Marius Kaminskas, Iván Cantador, Francesco Ricci. 2011. **A Generic Semantic-based Framework for Cross-domain Recommendation**. En *Actas de 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011)*, pp. 25-32. ACM, ISBN 978-1-4503-1027-7.

Otras publicaciones relacionadas con la tesis

El proyecto de la tesis fue defendido y evaluado en el Consorcio de Doctorado de 21st Conference on User Modeling, Adaptation, and Personalization:

Otras publicaciones

- Ignacio Fernández-Tobías. 2013. **Mining Semantic Data, User Generated Contents, and Contextual Information for Cross-Domain Recommendation**. En *Actas de 21st Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*, pp. 371-375. Lecture Notes in Computer Science 7899, Springer, ISBN 978-3-642-38843-9.

Relacionadas con el trabajo realizado sobre recomendación basada en etiquetado social, las siguientes publicaciones presentan técnicas de procesamiento y conjuntos de datos que se usaron posteriormente en los experimentos de la tesis:

- Iván Cantador, Alejandro Bellogín, Ignacio Fernández-Tobías, Sergio López-Hernández. 2011. **Semantic Contextualization of Social Tag-based Item Recommendations**. En *Actas de 12th International Conference on E-Commerce and Web Technologies (EC-Web 2011)*, pp. 101-113. Lecture Notes in Business Information Processing 85, Springer, ISBN 978-3-642-23013-4.

- Ignacio Fernández-Tobías, Iván Cantador, Alejandro Bellogín. 2011. **Semantic Disambiguation and Contextualisation of Social Tags**. En *Advances in User Modeling perspectives - selected papers from UMAP 2011 workshops*, pp. 181-197. Lecture Notes in Computer Science 7138, Springer, ISBN 978-3-642-28508-0.

- Ignacio Fernández-Tobías, Iván Cantador, Alejandro Bellogín. 2011. **cTag: Semantic Contextualisation of Social Tags**. En *Actas de 6th International Workshop on Semantic Adaptive Social Web (SASWeb 2011)*. CEUR Workshop Proceedings 730, ISSN 1613-0073.

B

CONCLUSIONES Y TRABAJO FUTURO

En esta tesis se ha abordado el problema del *arranque en frío* en filtrado colaborativo en un dominio objetivo por medio de recomendaciones sobre dominios cruzados, generadas mediante la expltación de diferentes tipos de datos auxiliares en un dominio de origen relacionado. Primero se ha propuesto una formalización del problema de recomendación sobre dominios cruzados que unifica las perspectivas de distintas áreas, junto con una categorización de aproximaciones previas atendiendo a sus procesos de agregación, enlazado, o transferencia de conocimiento entre dominios. Posteriormente, se han propuesto tres novedosas aproximaciones que extienden el modelo de factorización matricial para filtrado colaborativo sobre dominios cruzados, explotando etiquetas sociales, factores de personalidad de los usuarios, y metadatos semánticamente relacionados de los ítems. Finalmente, se han evaluado los models propuestos no sólo en términos de precisión, sino también considerando dimensiones alternativas de calidad de las recomendaciones, tales como novedad y diversidad de los ítems, y cobertura de los dominios.

En este capítulo se presentan las principales conclusiones derivadas de la investigación realizada. En la Sección B.1 se resumen el trabajo realizado en la tesis y se discuten las contribuciones resultantes, mientras que en la Sección B.2 se describen posibles temas de trabajo futuro.

B.1 RESUMEN Y DISCUSIÓN DE LAS CONTRIBUCIONES

En las siguientes subsecciones se resumen y discuten las principales contribuciones de esta tesis, abordando los objetivos de investigación planteados en el Apéndice A. Con respecto al objetivo de investigación O1, primero se realizó una revisión del estado del arte en recomendación sobre dominios cruzados, y se proporcionó una formalización unificadora del problema, así como una categorización de las propuestas existentes. En relación al objetivo O2, se desarrolló una serie de novedosos modelos de factorización matricial para filtrado colaborativo sobre dominios cruzados que explotan diferentes fuentes de información para enlazar los dominios de recomendación, abordando así el problema del arranque en frío en un dominio de destino. Finalmente, con respecto al objetivo de investigación O3, se evaluaron los modelos propuestos considerando múltiples aspectos de calidad de las recomendaciones, tales como la precisión, novedad y diversidad de los ítems, y la cobertura de los dominios.

B.1.1 *Formalización del problema de recomendación sobre dominios cruzados y categorización de las aproximaciones existentes*

Definición de dominio, tareas y objetivos en dominios cruzados

En el Capítulo 3 se realizó una novedosa y exhaustiva revisión del estado del arte en sistemas de recomendación sobre dominios cruzados. Se propuso una formalización del problema basada en una definición de dominio que considera varios niveles de granularidad, en particular atributos de ítem (e.g., géneros de películas), tipos de ítem (e.g., películas y series de televisión), ítems (e.g., películas y libros), y sistemas (e.g., MovieLens y Netflix). La definición propuesta generaliza nociones de dominio previas, y permite unificar anteriores perspectivas desde las que se ha abordado el problema. Además, se identificaron las principales tareas y objetivos de las aproximaciones para recomendación sobre dominios cruzados abordados en distintas áreas de investigación.

Agregación vs. vinculado y transferencia de conocimiento

En base a la técnica utilizada para enlazar dominios, se propuso una categorización de las aproximaciones para recomendación sobre dominios cruzados formada por dos grupos principales. El primero está compuesto por métodos que agregan conocimiento de diferentes dominios para realizar recomendaciones en un dominio de destino. En la literatura se identicaron diversos tipos de conocimiento agregado, a saber (i) preferencias de usuario, (ii) datos de modelado de usuario tales como similitudes y vecindarios, y (iii) listas de recomendación. El segundo grupo hace referencia a aproximaciones que vinculan y transfieren conocimiento entre dominios con el fin de soportar el proceso de recomendación. En particular, estas aproximaciones se clasificaron en (i) métodos que vinculan dominios utilizando conocimiento común, como redes semánticas y similitudes entre los dominios, (ii) aproximaciones que comparten factores latentes aprendidos en cada dominio, y (iii) aproximaciones que extraen y transfieren patrones de preferencias de usuario al dominio de destino.

Escenarios de solapamiento entre dominios y metodologías de evaluación

Además de la categorización previa, también se analizó el solapamiento entre dominios requerido por cada aproximacion, es decir, si se asume o no la existencia de usuarios o ítems comunes entre los dominios de origen y destino para generar recomendaciones. Finalmente, se resumió la variedad de metodologías que los diferentes trabajos han considerado hasta la fecha para evaluar recomendaciones sobre dominios cruzados. Se observó que la mayoría de los trabajos se han centrado en la tarea de predicción de *ratings*, y miden el desempeño de las aproximaciones propuestas en base a la precisión de las predicciones calculadas. Esto llevó a investigar en esta tesis modelos de recomendación sobre dominios cruzados para la tarea de ranking de ítems, tratando con preferencias de usuario sólo positivas en lugar de ratings numéricos.

B.1.2 *Modelos de recomendación sobre dominios cruzados para el arranque en frío*

Las principales contribuciones de esta tesis son tres novedosas aproximaciones basadas en factorización matricial para filtrado colaborativo sobre dominios cruzados, que explotan diferentes fuentes de información auxiliar para enlazar dominios y generar recomendaciones en el arranque en frío.

Factorización matricial basada en etiquetas sociales

En el Capítulo 4 se presentó el primer modelo de factorización matricial propuesto, que explota etiquetas sociales compartidas entre los dominios de interés para soportar la transferencia de conocimiento desde el dominio de origen al de destino. El modelo se basa en trabajo previo (Enrich et al., 2013) que utilizó etiquetas sociales para enlazar dominios y calcular predicciones de rating. Introduciendo un conjunto adicional de variables latentes, la aproximación propuesta modela separadamente la contribución de las etiquetas de usuarios e ítems, capturando así mejor su contribución a los ratings observados. La correlación entre las etiquetas de usuario/ítem y los ratings es transferida entre los dominios para mejorar las predicciones de rating de usuarios en arranque frío en el dominio de destino, incluso cuando éstos no hubiesen etiquetado el ítem correspondiente. Empíricamente se mostró que el modelo propuesto obtiene mayores mejoras con respecto a trabajos previos precisamente para usuarios con sólo unos pocos ratings en el dominio de destino.

Factorización matricial basada en personalidad de los usuarios

En lugar de explotar preferencias usuario-ítem expresadas por medio de etiquetas sociales para enlazar los dominios, en el Capítulo 5 se presentó una segunda aproximación que hace uso de información sobre la personalidad de los usuarios. En particular, se propusieron modelos de factorización matricial que aumentan los perfiles de usuario con variables latentes adicionales asociadas a atributos del *Modelo de Cinco Factores*, un modelo de representación de personalidad bien conocido en Psicología. Los modelos de factorización matricial basados en personalidad se diseñaron para tratar con preferencias de usuario sólo positivas, en vez de ratings numéricos. Con este propósito se desarrolló un procedimiento de mínimos cuadrados alternados para entrenar los modelos eficientemente. Las variables latentes adicionales correspondientes a factores de personalidad permiten la transferencia de conocimiento incluso cuando no hay solapamiento entre los dominios. En caso de que lo haya, los modelos propuestos se extendieron para explotar preferencias de usuario en dominios auxiliares. Los resultados alcanzados en los experimentos realizados indican que los modelos basados en personalidad propuestos son significativamente más efectivos que aproximaciones del estado del arte para usuarios completamente nuevos en el sistema. Sin embargo, se observa que una vez que se dispone de preferencias del usuario en el dominio de destino, el beneficio de explotar información de personalidad es únicamente marginal, cuando lo hay.

Factorización matricial basada en metadatos de ítem

Las dos aproximaciones previas utilizan respectivamente información de usuario-ítem en forma de etiquetas sociales e información de personalidad, específica de los usuarios. Los modelos propuestos en el Capítulo 6 hacen uso de metadatos semánticos de los ítems con el objetivo de enlazar los dominios para recomendación sobre dominios cruzados con preferencias sólo positivas. Basados en la hipótesis de que ítems semánticamente similares de diferentes dominios deberían tener factores latentes similares, se presentaron tres modelos de factorización matricial que explotan similitudes semánticas entre los dominios para regularizar el aprendizaje de los factores de ítem. Sin embargo, los metadatos a considerar además de las preferencias de los usuarios hacen que los modelos sean lentos. Por este motivo, para cada uno de los modelos propuestos, se adaptó un algoritmo de aprendizaje rápido para mínimos cuadrados alternados (Pilászy et al., 2010). Los resultados del trabajo experimental muestran que los modelos propuestos son capaces de proporcionar recomendaciones más precisas que métodos basados en grafos del estado del arte, dependiento de los dominios de recomendación considerados y de la cantidad de preferencias de usuario disponibles en el dominio de destino.

B.1.3 *Evaluación de precisión, novedad y diversidad de las recomendaciones, y cobertura de dominios*

Como se ha mencionado previamente, el análisis del estado del arte en recomendación sobre dominios cruzados muestra que la mayoría de las aproximaciones existentes han sido evaluadas midiendo el error de sus predicciones de ratings. En esta tesis se han considerado más dimensiones de calidad de las recomendaciones para comparar el desempeño de los modelos propuestos.

Evaluación de predicción de rating en el modelo de etiquetas sociales

Con el objetivo de comparar justamente el modelo de factorización matricial basado en etiquetas sociales, en el Capítulo 4 se trató de reproducir las condiciones y la metodología de evaluación seguidas en el trabajo previo (Enrich et al., 2013) sobre el que se basa el modelo propuesto. Por lo tanto, se evaluó el modelo propuesto en la misma tarea de predicción de rating, con el mismo conjunto de datos, y usando la misma métrica de error. Las aproximaciones evaluadas son capaces de transferir conocimiento entre los dominios incluso cuando no hay solapamiento de usuarios ni ítems, dado que sólo requieren solapamiento de etiquetas. Para simular niveles crecientes de arranque en frío, se siguió la metodología propuesta en (Enrich et al., 2013), que submuestrea aleatoriamente distintas cantidades de datos en el dominio de destino, lo que resulta en varios niveles de solapamiento de etiquetas. Los resultados obtenidos muestran que el modelo propuesto es más efectivo en condiciones severas de arranque en frío con menor solapamiento de etiquetas. También se analizó la precisión de las aproximaciones para usuarios con distinto número de ratings y etiquetas en el dominio de destino, y se observó que el modelo propuesto alcanza las mayores mejoras con respecto al estado del arte para usuarios con pocos ratings y etiquetas.

Evaluación de ranking de recomendaciones para los modelos basados en personalidad y metadatos

En el Capítulo 5 y en el Capítulo 6 se evaluaron los modelos basados en personalidad y metadatos utilizando distintos conjuntos de datos compuestos de preferencias de usuario sólo positivas formadas por *likes* de Facebook, en contrapartida a ratings numéricos. Como resultado, se midió el desempeño de los modelos en la tarea de ranking de ítems o recomendación de los N mejores, utilizando métricas de precisión como MAP y MRR en vez de métricas basadas en error como MAE y RMSE, tal como se había hecho previamente en la literatura. Más aún, se siguió una metodología fundamentada para situaciones de arranque en frío (Kluver y Konstan, 2014), que se extendió para tratar con dominios cruzados y para generar conjuntos de validación usados para afinar parámetros de los modelos y para evaluar estrategias de adquisición inteligente de preferencias.

Adicionalmente, en el Capítulo 5 se evaluó el modelo propuesto de factorización matricial basada en personalidad en dos casos, con y sin solapamiento. También se analizó la cobertura del catálogo de ítems recomendados, y la entropía de la distribución de los ítems como medidas de diversidad agregada. Los resultados empíricos alcanzados para el modelo basado en personalidad muestran que éste es capaz de proporcionar recomendaciones relevantes para usuarios recién llegados al sistema, y que los ítems sugeridos son significativamente diferentes que la recomendación de los más populares. Además, en el Capítulo 6 se dio un paso más extrayendo metadatos de los ítems, lo que permitió calcular similitudes entre los ítems recomendados y eventualmente medir la diversidad de las recomendaciones. En este contexto, se observó que a menudo hay un compromiso entre precisión y diversidad en condiciones de arranque en frío, y que aproximaciones con buen desempeño en términos de precisión tienen dificultades para proporcionar recomendaciones diversas en algunas situaciones. Por el contrario, en tales situaciones, las aproximaciones que proporcionan buena diversidad a menudo lo hacen recomendando ítems no relevantes.

B.2 TEMAS DE INVESTIGACIÓN ABIERTOS

Conveniencia de las fuentes de datos auxiliares

Como se ha mostrado en esta tesis, es posible explotar datos auxiliares de dominos de origen para proporcionar recomendaciones relevantes a nuevos usuarios en un dominio de destino. Sin embargo, en los experimentos realizados también se ha observado que la información adicional puede resultar no ser útil, e incluso perjudicial, cuando hay preferencias de usuario suficientes en el dominio de destino. A lo largo de esta tesis se ha observado este fenómeno *a posteriori* a través del análisis del desempeño de los modelos de recomendación en distintas situaciones de arranque en frío. En aplicaciones reales, por el contrario, un sistema debe decidir de antemano si la información del dominio de origen merece la pena o no ser explotada. Un mecanismo inteligente para determinar si las preferencias auxiliares deberían ser consideradas sería por tanto una componente deseable de un sistema de recomendación sobre dominios cruzados.

En Aprendizaje Automático se han propuesto tres aspectos diferentes a considerar en el desarrollo de modelos de transferencia de aprendizaje (Pan y Yang, 2010):

- *¿Qué transferir?* El primer aspecto está relacionado con el conocimiento específico a ser transferido entre los modelos. En esta tesis se han propuesto tres aproximaciones de factorización matricial que respectivamente transfieren factores latentes para etiquetas sociales, personalidad de los usuarios, y metadatos semánticos de los ítems.

- *¿Cómo transferir?* El segundo aspecto hace referencia al algoritmo de aprendizaje automático a utilizar para transferir el conocimiento. Los métodos propuestos en el Capítulo 4 y en el Capítulo 5 comparten vectores latentes de etiquetas sociales y factores de personalidad, respectivamente, y en el Capítulo 6 la transferencia se realiza mediante la regularización de factores de ítems con información de similitud durante el proceso de aprendizaje.

- *¿Cuándo transferir?* El tercer aspecto trata del problem de *transferencia negativa*, es decir, la situación en la que el conocimiento transferido no es útil para la tarea en cuestión. Una componente clave para el éxito de estas aproximaciones es decidir cuándo la información auxiliar debería ser transferida.

El tema de cuándo transferir conecta directamente con la observación previa en relación a la disponibilidad de preferencias en el dominio de destino. Así, como posible mejora de los modelos de factorización matricial propuestos en esta tesis, se plantea la aplicación de técnicas de aprendizaje automático para evitar automáticamente el problema de la *transferencia negativa*. En particular, los modelos probabilísticos que utilizan inferencia Bayesiana (Salakhutdinov y Mnih, 2008) suponen una buena solución para abordar el problema, dado que son capaces de calcular una medida de confianza en los parámetros que aprenden. Pocas preferencias de usuario en el dominio de destino resultarán en factores de usuario pobremente estimados con poca confianza, indicando que se debería explotar la información auxiliar del dominio de origen. Según comience a haber más preferencias disponibles, la incertidumbre decrecerá gradualmente hasta que los factores aprendidos sólo con datos del dominio de destino sean suficientemente precisos para calcular recomendaciones.

Procesado de etiquetas sociales

Con respecto a los modelos de recomendación basados en etiquetas sociales discutidos en el Capítulo 4, el trabajo previo de Enrich et al., (2013) mostró que descartar etiquetas no relevantes puede conllevar mejor precisión. El modelo TagGSVD++ propuesto, por el contrario, es capaz de superar a los métodos base modelando de forma separada el efecto de las etiquetas de usuarios e ítems. Sin embargo, podría ser posible obtener aún mejores resultados realizando una selección cuidadosa de las etiquetas más prometedoras. Además, en esta tesis no se realizó ningún preprocesado sofisticado de las etiquetas en los conjuntos de datos. Atendiendo a los trabajos de Szomszor et al., (2008a) y Abel et al., (2011), se plantea que alinear las etiquetas de dominios distintos en base a sus significados puede ser útil para aumentar el solapamiento entre los dominios y, eventualmente, la calidad de las recomendaciones.

Representación de la personalidad del usuaro

En el Capítulo 5 se aumentó el modelo de factorización matricial con atributos que representan valores de personalidad según el Modelo de Cinco Factores, una representación bien conocida en Psicología que considera cinco factores ortogonales de la personalidad. Se probaron varias estrategias de discretización para transformar los valores de personalidad en variables para los modelos de recomendación propuestos, y los resultados alcanzados muestran que la información de personalidad es útil para usuarios nuevos de los que el sitema no dispone ninguna preferencia. Sin embargo, es posible que representaciones de personalidad alternativas correlacionen mejor con las preferencias de los usuarios, y sean por tanto más adecuados para la recomendación. Por ejemplo, se podrían obtener mejores resultados utilizando representaciones más precisas basadas en facetas de personalidad, tales como la *imaginación* o el grado de *interés artístico*, ambas asociadas al factor de *apertura al cambio* (*openness* en inglés). Además, durante el Capítulo 5 se asumió que la información de personalidad estaba siempre disponible. En aplicaciones reales es necesario obtener primero dicha información, lo cual normalmente se hace pidiendo a los usuarios que rellenen cuestionarios a partir de los cuales se estiman los parámetros del Modelo de Cinco Factores. Esta es una importante limitación en situaciones de arranque en frío, en las que los usuarios acaban de registrarse en el sistema esperando recibir recomendaciones, y pueden no estar dispuestos a emplear tiempo proporcionando dicha información. En este contexto, una posible solución es obtener los datos de personalidad implícitamente por medio del análisis de patrones de comportamiento durante la interacción del usuario con otros y con el sistema (Kosinski et al., 2013).

Selección de características semánticas

Con respecto al Capítulo 6, los modelos de factorización matricial para dominios cruzados que explotan metadatos semánticos están basados en similitudes entre dominios calculadas utilizando la estructura de enlaces de las redes semánticas en Wikipedia/DBpedia. Dichas similitudes, sin embargo, no tienen en cuenta la relevancia de las propiedades semánticas que conectan los dominios de interés. Trabajo reciente (Musto et al., 2016b) ha mostrado que una selección cuidadosa de las propiedades más relevantes puede conllevar mejor desempeño al tanto que reduce el tamaño de las redes semánticas, lo que resulta en una mayor eficiencia de los algoritmos de recomendación.

Estudios con usuarios

Finalmente, se comenta que todo el trabajo experimental de esta tesis se ha realizado en condiciones *offline*, utilizando conjuntos de datos que contienen preferencias de usuario en varios dominios. Hasta lo mejor de nuestro conocimiento, no existe ningún conjunto de datos público que contenga al mismo tiempo preferencias de usuario y los tres tipos de datos considerados en esta tesis: etiquetas sociales, factores de personalidad, y metadatos semánticos. Sería interesante determinar cuál de las fuentes de datos consideradas es más valiosa para proporcionar recomendaciones sobre dominios cruzados en situaciones de arranque en frío, pero la falta de dicho conjunto de datos imposibilita la comparación entre sí de los modelos propuestos en la tesis. Una posible dirección de trabajo futuro es por tanto realizar un estudio con usuarios recogiendo dicha información de un grupo lo suficientemente grande, solicitando su valoración de las recomendaciones generadas por cada modelo.

REFERENCES

Abel, F., Araújo, S., Gao, Q., and Houben, G.-J. (2011). Analyzing Cross-System User Modeling on the Social Web. In: *Proceedings of the 11th International Conference on Web Engineering (ICWE 2011)*. Vol. 6757, pp. 28–43.

Abel, F., Herder, E., Houben, G.-J., Henze, N., and Krause, D. (2013). Cross-system user modeling and personalization on the Social Web. In: *User Modeling and User-Adapted Interaction* 23 (2-3), pp. 169–209.

Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. In: *IEEE Transactions on Knowledge and Data Engineering* 17 (6), pp. 734–749.

Adomavicius, G. and Tuzhilin, A. (2015). Context-Aware Recommender Systems. In: *Recommender Systems Handbook*, pp. 191–226.

Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. In: *ACM Transactions on Information Systems* 23 (1), pp. 103–145.

Azak, M. (2010). Crossing: A Framework to Develop Knowledge-based Recommenders in Cross Domains. MSc thesis. Middle East Technical University.

Bachrach, Y., Kosinski, M., Graepel, T., Kohli, P., and Stillwell, D. (2012). Personality and patterns of Facebook usage. In: *Proceedings of the 3rd Annual ACM Web Science Conference*. ACM, pp. 24–32.

Baeza-Yates, R. A. and Ribeiro-Neto, B. A. (1999). *Modern Information Retrieval*. ACM Press / Addison-Wesley.

Baltrunas, L. and Ricci, F. (2009). Context-based splitting of item ratings in collaborative filtering. In: *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, pp. 245–248.

Baltrunas, L., Ludwig, B., and Ricci, F. (2011). Matrix factorization techniques for context aware recommendation. In: *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*, pp. 301–304.

Barjasteh, I., Forsati, R., Masrour, F., Esfahanian, A.-H., and Radha, H. (2015). Cold-Start Item and User Recommendation with Decoupled Completion and Transduction. In: *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys 2015)*, pp. 91–98.

Bell, R. M. and Koren, Y. (2007a). Lessons from the Netflix prize challenge. In: *SIGKDD Explorations* 9 (2), pp. 75–79.

Bell, R. M. and Koren, Y. (2007b). Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights. In: *Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007)*, pp. 43–52.

Bellogín, A. and Parapar, J. (2012). Using graph partitioning techniques for neighbour selection in user-based collaborative filtering. In: *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys 2012)*, pp. 213–216.

Bellogín, A., Cantador, I., Díez, F., Castells, P., and Chavarriaga, E. (2013). An Empirical Comparison of Social, Collaborative Filtering, and Hybrid Recommenders. In: *ACM Transactions on Intelligent Systems and Technology* 4 (1), pp. 1–29.

Berkovsky, S., Kuflik, T., and Ricci, F. (2005). Entertainment Personalization Mechanism Through Cross-Domain User Modeling. In: *Proceedings of the 1st International Conference on Intelligent Technologies for Interactive Entertainment*. Vol. 3814, pp. 215–219.

Berkovsky, S., Goldwasser, D., Kuflik, T., and Ricci, F. (2006). Identifying Inter-Domain Similarities Through Content-Based Analysis of Hierarchical Web-Directories. In: *Proceedings of the 17th European Conference on Artificial Intelligence*. Vol. 141, pp. 789–790.

Berkovsky, S., Kuflik, T., and Ricci, F. (2007a). Cross-Domain Mediation in Collaborative Filtering. In: *Proceedings of the 11th International Conference on User Modeling*. Vol. 4511, pp. 355–359.

Berkovsky, S., Kuflik, T., and Ricci, F. (2007b). Distributed collaborative filtering with domain specialization. In: *Proceedings of the 1st ACM Conference on Recommender Systems (RecSys 2007)*, pp. 33–40.

Berkovsky, S., Kuflik, T., and Ricci, F. (2008). Mediation of user models for enhanced personalization in recommender systems. In: *User Modeling and User-Adapted Interaction* 18 (3), pp. 245–286.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked Data - The Story So Far. In: *International Journal on Semantic Web and Information Systems* 5 (3), pp. 1–22.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. In: *Journal of Machine Learning Research* 3, pp. 993–1022.

Braunhofer, M., Kaminskas, M., and Ricci, F. (2013). Location-aware music recommendation. In: *International Journal of Multimedia Information Retrieval* 2 (1), pp. 31–44.

Braunhofer, M., Ricci, F., Lamche, B., and Wörndl, W. (2015a). A Context-Aware Model for Proactive Recommender Systems in the Tourism Domain. In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (MobileHCI 2015)*, pp. 1070–1075.

Braunhofer, M., Fernández-Tobías, I., and Ricci, F. (2015b). Parsimonious and Adaptive Contextual Information Acquisition in Recommender Systems. In: *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS 2015)*. Vol. 1438, pp. 2–8.

Braunhofer, M., Elahi, M., and Ricci, F. (2015c). User Personality and the New User Problem in a Context-Aware Point of Interest Recommender System. In: *Information and Communication Technologies in Tourism 2015*, pp. 537–549.

Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In: *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 43–52.

Burger, J. M. (2010). *Personality*. Wadsworth Publishing.

Campos, L. M. de, Fernández-Luna, J. M., Huete, J. F., and Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. In: *International Journal of Approximate Reasoning* 51 (7), pp. 785–799.

Campos, P. G., Díez, F., and Cantador, I. (2014). Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. In: *User Modeling and User-Adapted Interaction* 24 (1-2), pp. 67–119.

Cantador, I., Konstas, I., and Jose, J. M. (2010a). Categorising social tags to improve folksonomy-based recommendations. In: *Journal of Web Semantics* 9 (1), pp. 1–15.

Cantador, I., Bellogín, A., and Vallet, D. (2010b). Content-based recommendation in social tagging systems. In: *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys 2010)*, pp. 237–240.

Cantador, I., Fernández-Tobías, I., and Bellogín, A. (2013). Relating Personality Types with User Preferences in Multiple Entertainment Domains. In: *1st International Workshop on Emotions and Personality in Personalized Services, CEUR workshop Proceedings*. Vol. 997.

Cantador, I., Fernández-Tobías, I., Berkovsky, S., and Cremonesi, P. (2015). Cross-Domain Recommender Systems. In: *Recommender Systems Handbook*, pp. 919–959.

Cao, B., Liu, N. N., and Yang, Q. (2010). Transfer Learning for Collective Link Prediction in Multiple Heterogenous Domains. In: *Proceedings of the 27th International Conference on Machine Learning (ICML 2010)*, pp. 159–166.

Carmagnola, F. and Cena, F. (2009). User identification for cross-system personalisation. In: *Information Sciences* 179 (1-2), pp. 16–32.

Carmagnola, F., Cena, F., and Gena, C. (2011). User model interoperability: a survey. In: *User Modeling and User-Adapted Interaction* 21 (3), pp. 285–331.

Castells, P., Hurley, N. J., and Vargas, S. (2015). Novelty and Diversity in Recommender Systems. In: *Recommender Systems Handbook*, pp. 881–918.

Celma, O. (2010). Music Recommendation and Discovery in the Long Tail. PhD thesis. Universitat Pompeu Fabra, Barcelona, Spain.

Celma, Ò. and Herrera, P. (2008). A new approach to evaluating novel recommendations. In: *Proceedings of the 2nd ACM Conference on Recommender Systems (RecSys 2008)*, pp. 179–186.

Chausson, O. (2010). *Who watches what? Assessing the impact of gender and personality on film preferences*.

Chung, R., Sundaram, D., and Srinivasan, A. (2007). Integrated personal recommender systems. In: *Proceedings of the 9th International Conference on Electronic Commerce: The Wireless World of Electronic Commerce*. Vol. 258, pp. 65–74.

Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., and Sartin, M. (1999). Combining content-based and collaborative filters in an online newspaper. In: *Proceedings of ACM SIGIR Workshop on Recommender Systems*. Vol. 60. Citeseer.

Codina, V., Ricci, F., and Ceccaroni, L. (2013). Exploiting the Semantic Similarity of Contextual Situations for Pre-filtering Recommendation. In: *Proceedings of the 21st International Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*. Vol. 7899, pp. 165–177.

Costa, P. T. and McCrae, R. R. (1992). *Revised NEO Personality Inventory (NEO PI-R) and NEO Five-Factor Inventory (NEO FFI): Professional Manual*. Psychological Assessment Resources.

Cremonesi, P. and Quadrana, M. (2014). Cross-domain recommendations without overlapping data: myth or reality? In: *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, pp. 297–300.

Cremonesi, P., Koren, Y., and Turrin, R. (2010). Performance of recommender algorithms on top-n recommendation tasks. In: *Proceed-*

ings of the 4th ACM Conference on Recommender Systems (RecSys 2010), pp. 39–46.

Cremonesi, P., Garzotto, F., Negro, S., Papadopoulos, A. V., and Turrin, R. (2011a). Comparative evaluation of recommender system quality. In: *Proceedings of the International Conference on Human Factors in Computing Systems (CHI 2011)*, pp. 1927–1932.

Cremonesi, P., Tripodi, A., and Turrin, R. (2011b). Cross-Domain Recommender Systems. In: *IEEE 11th International Conference on Data Mining Workshops (ICDMW)*, pp. 496–503.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by Latent Semantic Analysis. In: *Journal of the American Society for Information Science* 41 (6), pp. 391–407.

Di Noia, T., Ostuni, V. C., Tomeo, P., and Di Sciascio, E. (2016). SPrank: Semantic Path-Based Ranking for Top-N Recommendations Using Linked Open Data. In: *ACM Transactions on Intelligent Systems and Technology* 8 (1), p. 9.

Ding, C. H. Q., Li, T., Peng, W., and Park, H. (2006). Orthogonal Nonnegative Matrix Tri-factorizations for Clustering. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 126–135.

Driskill, R. and Riedl, J. (1999). Recommender Systems for E-Commerce: Challenges and Opportunities. In: *AAAI '99 Workshop on Artifical Intelligence for Electronic Commerce*, pp. 73–76.

Ekstrand, M. D., Harper, F. M., Willemsen, M. C., and Konstan, J. A. (2014). User perception of differences in recommender algorithms. In: *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, pp. 161–168.

Elahi, M., Braunhofer, M., Ricci, F., and Tkalcic, M. (2013). Personality-based active learning for collaborative filtering recommender systems. In: *The 12th Symposium of the Italian Association for Artificial Intelligence*. Vol. 8249, pp. 360–371.

Elahi, M., Ricci, F., and Rubens, N. (2014). Active learning strategies for rating elicitation in collaborative filtering: A system-wide perspective. In: *ACM Transactions on Interactive Intelligent Systems* 5 (1), 13:1–13:33.

Elahi, M., Ricci, F., and Rubens, N. (2016). A survey of active learning in collaborative filtering recommender systems. In: *Computer Science Review* 20, pp. 29–50.

Enrich, M., Braunhofer, M., and Ricci, F. (2013). Cold-Start Management with Cross-Domain Collaborative Filtering and Tags. In:

Proceedings of the 14th International Conference on E-Commerce and Web Technologies (EC-Web 2013). Vol. 152, pp. 101–112.

Farnadi, G., Sitaraman, G., Sushmita, S., Celli, F., Kosinski, M., Stillwell, D., Davalos, S., Moens, M.-F., and Cock, M. D. (2016). Computational personality recognition in social media. In: *User Modeling and User-Adapted Interaction* 26 (2-3), pp. 109–142.

Fernández-Tobías, I. and Cantador, I. (2014). Exploiting Social Tags in Matrix Factorization Models for Cross-domain Collaborative Filtering. In: *Proceedings of the 1st Workshop on New Trends in Content-based Recommender Systems*. Vol. 1245, pp. 34–41.

Fernández-Tobías, I. and Cantador, I. (2015). On the Use of Cross-Domain User Preferences and Personality Traits in Collaborative Filtering. In: *Proceedings of the 23rd International Conference on User Modeling, Adaption and Personalization (UMAP 2015)*. Vol. 9146, pp. 343–349.

Fernández-Tobías, I., Cantador, I., Kaminskas, M., and Ricci, F. (2011). A Generic Semantic-based Framework for Cross-domain Recommendation. In: *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, pp. 25–32.

Fernández-Tobías, I., Cantador, I., Kaminskas, M., and Ricci, F. (2012). Cross-domain Recommender Systems: A Survey of the State of the Art. In: *Proceedings of the 2nd Spanish Conference on Information Retrieval*, pp. 187–198.

Fernández-Tobías, I., Cantador, I., and Plaza, L. (2013). An Emotion Dimensional Model Based on Social Tags: Crossing Folksonomies and Enhancing Recommendations. In: *Proceedings of the 14th International Conference on E-Commerce and Web Technologies (EC-Web 2013)*. Vol. 152, pp. 88–100.

Funk, S. (2006). *Netflix Update: Try This At Home*. URL: http://sifter.org/~simon/journal/20061211.html.

Gabrilovich, E. and Markovitch, S. (2007). Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. In: *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, pp. 1606–1611.

Gao, S., Luo, H., Chen, D., Li, S., Gallinari, P., and Guo, J. (2013). Cross-Domain Recommendation via Cluster-Level Latent Factor Model. In: *Proceedings of the 2013 European Conference on Machine Learning and Knowledge Discovery in Databases*. Vol. 8189, pp. 161–176.

Gemmell, J., Schimoler, T., Mobasher, B., and Burke, R. D. (2012). Resource recommendation in social annotation systems: A linear-weighted hybrid approach. In: *Journal of Computer and System Sciences* 78 (4), pp. 1160–1174.

Givon, S. and Lavrenko, V. (2009). Predicting social-tags for cold start book recommendations. In: *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, pp. 333–336.

Goga, O., Lei, H., Parthasarathi, S. H. K., Friedland, G., Sommer, R., and Teixeira, R. (2013). Exploiting innocuous activity for correlating users across sites. In: *Proceedings of the 22nd International World Wide Web Conference (WWW 2013)*, pp. 447–458.

Goldberg, L. R., Johnson, J. A., Eber, H. W., Hogan, R., Ashton, M. C., Cloninger, C. R., and Gough, H. G. (2006). The international Personality Item Pool and the Future of Public-Domain Personality Measures. In: *Journal of Research in Personality* 40 (1), pp. 84–96.

Golder, S. A. and Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. In: *Journal of Information Science* 32 (2), pp. 198–208.

González, G., López, B., and Rosa, J. L. de la (2005). A Multi-agent Smart User Model for Cross-domain Recommender Systems. In: *Beyond Personalization 2005 - The Next Stage of Recommender Systems Research*, pp. 93–94.

Gunawardana, A. and Shani, G. (2015). Evaluating Recommender Systems. In: *Recommender Systems Handbook*, pp. 265–308.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer-Verlag New York.

Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999). An Algorithmic Framework for Performing Collaborative Filtering. In: *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1999)*, pp. 230–237.

Hofmann, T. (2004). Latent semantic models for collaborative filtering. In: *ACM Transactions on Information Systems* 22 (1), pp. 89–115.

Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. (2006). Information Retrieval in Folksonomies: Search and Ranking. In: *Proceedings of the 3rd European Semantic Web Conference (ESWC 2006)*. Vol. 4011, pp. 411–426.

Hu, L., Cao, J., Xu, G., Cao, L., Gu, Z., and Zhu, C. (2013). Personalized Recommendation via Cross-Domain Triadic Factorization. In: *Proceedings of the 22nd International World Wide Web Conference (WWW 2013)*, pp. 595–605.

Hu, R. and Pu, P. (2009). A comparative user study on rating vs. personality quiz based preference elicitation methods. In: *Proceed-*

ings of the 14th International Conference on Intelligent User Interfaces, pp. 367–372.

Hu, R. and Pu, P. (2011). Enhancing collaborative filtering systems with personality information. In: *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*, pp. 197–204.

Hu, Y., Koren, Y., and Volinsky, C. (2008). Collaborative Filtering for Implicit Feedback Datasets. In: *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008)*, pp. 263–272.

Hulpus, I., Prangnawarat, N., and Hayes, C. (2015). Path-Based Semantic Relatedness on Linked Data and Its Use to Word and Entity Disambiguation. In: *Proceedings of the 14th International Semantic Web Conference (ISWC 2015)*. Vol. 9366, pp. 442–457.

Jain, P., Kumaraguru, P., and Joshi, A. (2013). @i seek 'fb.me': identifying users across multiple online social networks. In: *Proceedings of the 22nd International World Wide Web Conference (WWW 2013), Companion Volume*, pp. 1259–1268.

Jamali, M. and Ester, M. (2010). A matrix factorization technique with trust propagation for recommendation in social networks. In: *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys 2010)*, pp. 135–142.

Jannach, D. and Adomavicius, G. (2016). Recommendations with a Purpose. In: *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys 2016)*, pp. 7–10.

Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. In: *ACM Transactions on Information Systems* 20 (4), pp. 422–446.

John, O. P. and Srivastava, S. (1999). *The Big Five trait taxonomy: History, measurement, and theoretical perspectives.* Handbook of personality: Theory and research.

Kaminskas, M., Fernández-Tobías, I., Ricci, F., and Cantador, I. (2013). Ontology-based Identification of Music for Places. In: *Proceedings of the 13th International Conference on Information Communication Technologies in Tourism*, pp. 436–447.

Kaminskas, M., Fernández-Tobías, I., Ricci, F., and Cantador, I. (2014). Knowledge-based identification of music suited for places of interest. In: *Information Technology & Tourism* 14 (1), pp. 73–95.

Karatzoglou, A., Baltrunas, L., and Shi, Y. (2013). Learning to rank for recommender systems. In: *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys 2013)*, pp. 493–494.

Kitts, B., Freed, D., and Vrieze, M. (2000). Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities. In: *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 437–446.

Kluver, D. and Konstan, J. A. (2014). Evaluating recommender behavior for new users. In: *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, pp. 121–128.

Knijnenburg, B. P. and Willemsen, M. C. (2015). Evaluating Recommender Systems with User Experiments. In: *Recommender Systems Handbook*, pp. 309–352.

Kohavi, R., Longbotham, R., Sommerfield, D., and Henne, R. M. (2009). Controlled experiments on the web: survey and practical guide. In: *Data Mining and Knowledge Discovery* 18 (1), pp. 140–181.

Kook, H. J. (2005). Profiling Multiple Domains of User Interests and Using Them for Personalized Web Support. In: *Proceedings of the 1st International Conference on Intelligent Computing (ICIC 2005)*. Vol. 3645, pp. 512–520.

Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 426–434.

Koren, Y. and Bell, R. M. (2015). Advances in Collaborative Filtering. In: *Recommender Systems Handbook*, pp. 77–118.

Koren, Y., Bell, R. M., and Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. In: *IEEE Computer* 42 (8), pp. 30–37.

Kosinski, M, Stillwell, D, and Graepel, T (2013). Private traits and attributes are predictable from digital records of human behavior. In: *Proceedings of the National Academy of Sciences*, pp. 2–5.

Kosinski, M., Stillwell, D., Kohli, P., Bachrach, Y., and Graepel, T. (2012). Personality and Website Choice. In: *Proceedings of the 3rd Annual ACM Web Science Conference*.

Lawrence, N. D. and Urtasun, R. (2009). Non-linear matrix factorization with Gaussian processes. In: *Proceedings of the 26th International Conference on Machine Learning (ICML 2009)*. Vol. 382, pp. 601–608.

Lee, C.-H., Kim, Y.-H., and Rhee, P.-K. (2001). Web personalization expert with combining collaborative filtering and association rule mining technique. In: *Expert Systems with Applications* 21 (3), pp. 131–137.

Lee, D. D. and Seung, H. S. (2001). Algorithms for Non-negative Matrix Factorization. In: *Advances in Neural Information Processing Systems 13*, pp. 556–562.

Lehmann, J. et al. (2015). DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. In: *Semantic Web Journal* 6 (2), pp. 167–195.

Lepri, B., Staiano, J., Shmueli, E., Pianesi, F., and Pentland, A. (2016). The role of personality in shaping social networks and mediating behavioral change. In: *User Modeling and User-Adapted Interaction* 26 (2-3), pp. 143–175.

Li, B. (2011). Cross-Domain Collaborative Filtering: A Brief Survey. In: *IEEE 23rd International Conference on Tools with Artificial Intelligence (ICTAI 2011)*, pp. 1085–1086.

Li, B., Yang, Q., and Xue, X. (2009a). Can Movies and Books Collaborate? Cross-Domain Collaborative Filtering for Sparsity Reduction. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pp. 2052–2057.

Li, B., Yang, Q., and Xue, X. (2009b). Transfer learning for collaborative filtering via a rating-matrix generative model. In: *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*. Vol. 382, pp. 617–624.

Li, B., Zhu, X., Li, R., Zhang, C., Xue, X., and Wu, X. (2011). Cross-Domain Collaborative Filtering over Time. In: *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pp. 2293–2298.

Linden, G., Smith, B., and York, J. (2003). Amazon.com Recommendations: Item-to-Item Collaborative Filtering. In: *IEEE Internet Computing* 7 (1), pp. 76–80.

Liu, F. and Lee, H. J. (2010). Use of social network information to enhance collaborative filtering performance. In: *Expert Systems with Applications* 37 (7), pp. 4772–4778.

Loizou, A. (2009). How to Recommend Music to Film Buffs: Enabling the Provision of Recommendations from Multiple Domains. PhD thesis. University of Southampton.

Loni, B., Shi, Y., Larson, M., and Hanjalic, A. (2014). Cross-Domain Collaborative Filtering with Factorization Machines. In: *Proceedings of the 36th European Conference on IR Research on Advances in Information Retrieval (ECIR 2014)*. Vol. 8416, pp. 656–661.

Lops, P., Gemmis, M. de, and Semeraro, G. (2011). Content-based Recommender Systems: State of the Art and Trends. In: *Recommender Systems Handbook*, pp. 73–105.

Low, Y., Agarwal, D., and Smola, A. J. (2011). Multiple domain user personalization. In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 123–131.

Ma, H., King, I., and Lyu, M. R. (2007). Effective missing data prediction for collaborative filtering. In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39–46.

Ma, H., Yang, H., Lyu, M. R., and King, I. (2008). SoRec: social recommendation using probabilistic matrix factorization. In: *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM 2008)*, pp. 931–940.

Ma, H., King, I., and Lyu, M. R. (2009). Learning to recommend with social trust ensemble. In: *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2009)*, pp. 203–210.

Manzato, M. G. (2013). gSVD++: supporting implicit feedback on recommender systems with metadata awareness. In: *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC 2013)*, pp. 908–913.

Marinho, L. B., Nanopoulos, A., Schmidt-Thieme, L., Jäschke, R., Hotho, A., Stumme, G., and Symeonidis, P. (2011). Social Tagging Recommender Systems. In: *Recommender Systems Handbook*, pp. 615–644.

Marlin, B. M. and Zemel, R. S. (2009). Collaborative prediction and ranking with non-random missing data. In: *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, pp. 5–12.

McNee, S. M., Riedl, J., and Konstan, J. A. (2006). Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: *Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems*, pp. 1097–1101.

Milne, D. and Witten, I. H. (2008). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In: *Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, pp. 25–30.

Moreno, O., Shapira, B., Rokach, L., and Shani, G. (2012). TALMUD: transfer learning for multiple domains. In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM 2012)*, pp. 425–434.

Musto, C., Narducci, F., Lops, P., Gemmis, M. de, and Semeraro, G. (2016a). ExpLOD: A Framework for Explaining Recommendations based on the Linked Open Data Cloud. In: *Proceedings* *of the 10th ACM Conference on Recommender Systems (RecSys 2016)*, pp. 151–154.

Musto, C., Lops, P., Basile, P., Gemmis, M. de, and Semeraro, G. (2016b). Semantics-aware Graph-based Recommender Systems Exploiting Linked Open Data. In: *Proceedings of the 24th Conference on User Modeling Adaptation and Personalization, (UMAP 2016)*, pp. 229–237.

Nakatsuji, M., Fujiwara, Y., Tanaka, A., Uchiyama, T., and Ishida, T. (2010). Recommendations Over Domain Specific User Graphs. In: *Proceedings of the 19th European Conference on Artificial Intelligence*. Vol. 215, pp. 607–612.

Ning, X., Desrosiers, C., and Karypis, G. (2015). A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In: *Recommender Systems Handbook*, pp. 37–76.

Noll, M. G. and Meinel, C. (2007). Web Search Personalization Via Social Bookmarking and Tagging. In: *Proceedings of the 6th International Semantic Web Conference (ISWC 2007)*. Vol. 4825, pp. 367–380.

Nunes, M. A. S. N. (2009). *Recommender Systems based on Personality Traits: Could Human Psychological Aspects Influence the Computer Decision-making Process?* VDM Verlag.

Nunes, M. A. S. N. and Hu, R. (2012). Personality-based Recommender Systems: An Overview. In: *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys 2012)*, pp. 5–6.

Odic, A., Tkalcic, M., Tasic, J. F., and Kosir, A. (2013). Personality and social context: Impact on emotion induction from movies. In: *Late-Breaking Results, Project Papers and Workshop Proceedings of the 21st Conference on User Modeling, Adaptation, and Personalization*.

Oku, K., Nakajima, S., Miyazaki, J., and Uemura, S. (2006). Context-Aware SVM for Context-Dependent Information Recommendation. In: *Proceedings of the 7th International Conference on Mobile Data Management (MDM 2006)*, p. 109.

Pan, S. J. and Yang, Q. (2010). A Survey on Transfer Learning. In: *IEEE Transactions on Knowledge and Data Engineering* 22 (10), pp. 1345–1359.

Pan, S. J., Kwok, J. T., and Yang, Q. (2008). Transfer Learning via Dimensionality Reduction. In: *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 2008)*, pp. 677–682.

Pan, W. and Yang, Q. (2013). Transfer learning in heterogeneous collaborative filtering domains. In: *Artificial Intelligence* 197, pp. 39–55.

Pan, W., Xiang, E. W., Liu, N. N., and Yang, Q. (2010). Transfer Learning in Collaborative Filtering for Sparsity Reduction. In: *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI 2010)*, pp. 210–235.

Pan, W., Liu, N. N., Xiang, E. W., and Yang, Q. (2011). Transfer Learning to Predict Missing Ratings via Heterogeneous User Feedbacks. In: *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pp. 2318–2323.

Pan, W., Xiang, E. W., and Yang, Q. (2012). Transfer Learning in Collaborative Filtering with Uncertain Ratings. In: *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pp. 662–668.

Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., and Pedone, A. (2009). Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems. In: *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, pp. 265–268.

Paquet, U. and Koenigstein, N. (2013). One-class collaborative filtering with random graphs. In: *Proceedings of the 22nd International World Wide Web Conference (WWW 203)*, pp. 999–1008.

Parra, D., Karatzoglou, A., Yavuz, I., and Amatriain, X. (2011). Implicit feedback recommendation via implicit-to-explicit ordinal logistic regression mapping. In: *In Proceedings of the 3rd Workshop on Context-Aware Recommender Systems (CARS 2011)*.

Passant, A. (2010). dbrec - Music Recommendations Using DBpedia. In: *Proceedings of the 9th International Semantic Web Conference (ISWC 2010)*. Vol. 6497, pp. 209–224.

Pazzani, M. J. (1999). A Framework for Collaborative, Content-Based and Demographic Filtering. In: *Artificial Intelligence Review* 13 (5-6), pp. 393–408.

Pazzani, M. J. and Billsus, D. (1997). Learning and Revising User Profiles: The Identification of Interesting Web Sites. In: *Machine Learning* 27 (3), pp. 313–331.

Pilászy, I., Zibriczky, D., and Tikk, D. (2010). Fast als-based matrix factorization for explicit and implicit feedback datasets. In: *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys 2010)*, pp. 71–78.

Rawlings, D. and Ciancarelli, V. (1997). Music preference and the five-factor model of the NEO Personality Inventory. In: *Psychology of Music* 25 (2), pp. 120–132.

Rendle, S. (2012). Factorization Machines with libFM. In: *ACM Transactions on Intelligent Systems and Technology* 3 (3), p. 57.

Rentfrow, P. J., Gosling, S. D., and Others (2003). The Do Re Mi's of everyday life: The structure and personality correlates of music preferences. In: *Journal of Personality and Social Psychology* 84 (6), pp. 1236–1256.

Rentfrow, P. J., Goldberg, L. R., and Zilca, R. (2011). Listening, watching, and reading: The structure and correlates of entertainment preferences. In: *Journal of Personality* 79 (2), pp. 223–258.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In: *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pp. 175–186.

Roshchina, A. (2012). *TWIN Personality-based Recommender System*. Institute of Technology Tallaght, Dublin.

Rowe, M. (2014). SemanticSVD++: Incorporating Semantic Taste Evolution for Predicting Ratings. In: *IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, pp. 213–220.

Sahebi, S. and Brusilovsky, P. (2013). Cross-Domain Collaborative Recommendation in a Cold-Start Context: The Impact of User Profile Size on the Quality of Recommendation. In: *Proceedings of the 21st International Conference on User Modeling, Adaptation, and Personalization (UMAP 2013)*. Vol. 7899, pp. 289–295.

Salakhutdinov, R. and Mnih, A. (2007). Probabilistic Matrix Factorization. In: *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pp. 1257–1264.

Salakhutdinov, R. and Mnih, A. (2008). Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In: *Proceedings of the 25th International Conference on Machine Learning (ICML 2008)*. Vol. 307, pp. 880–887.

Salakhutdinov, R., Mnih, A., and Hinton, G. E. (2007). Restricted Boltzmann machines for collaborative filtering. In: *Proceedings of the 24th International Conference on Machine Learning (ICML 2007)*. Vol. 227, pp. 791–798.

Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In: *Proceedings of the 10th International World Wide Web Conference*, pp. 285–295.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2000). Application of Dimensionality Reduction in Recommender System - A Case Study. In: *ACM WebKDD 2000 Workshop*. ACM SIGKDD.

Sen, S., Vig, J., and Riedl, J. (2009). Tagommenders: connecting users to items through tags. In: *Proceedings of the 18th International World Wide Web Conference (WWW 2009)*, pp. 671–680.

Shadbolt, N., Berners-Lee, T., and Hall, W. (2006). The Semantic Web Revisited. In: *IEEE Intelligent Systems* 21 (3), pp. 96–101.

Shapira, B., Rokach, L., and Freilikhman, S. (2013). Facebook single and cross domain data for recommendation systems. In: *User Modeling and User-Adapted Interaction* 23 (2-3), pp. 211–247.

Shi, C., Kong, X., Huang, Y., Yu, P. S., and Wu, B. (2014). HeteSim: A General Framework for Relevance Measure in Heterogeneous Networks. In: *IEEE Transactions on Knowledge and Data Engineering* 26 (10), pp. 2479–2492.

Shi, Y., Larson, M., and Hanjalic, A. (2011). Tags as Bridges between Domains: Improving Recommendation with Tag-Induced Cross-Domain Collaborative Filtering. In: *Proceedings of the 19th International Conference on User Modeling, Adaption and Personalization (UMAP 2011)*. Vol. 6787, pp. 305–316.

Shi, Y., Karatzoglou, A., Baltrunas, L., Larson, M., Oliver, N., and Hanjalic, A. (2012). CLiMF: learning to maximize reciprocal rank with collaborative less-is-more filtering. In: *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys 2012)*, pp. 139–146.

Smyth, B. and McClave, P. (2001). Similarity vs. Diversity. In: *Proceedings of the 4th International Conference on Case-Based Reasoning (ICCBR 2001)*. Vol. 2080, pp. 347–361.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian Optimization of Machine Learning Algorithms. In: *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS 2012)*, pp. 2960–2968.

Soboroff, I. and Nicholas, C. (1999). Combining content and collaboration in text filtering. In: *Proceedings of the IJCAI '99 Workshop on Machine Learning for Information Filtering*. Vol. 99. Citeseer, pp. 86–91.

Steck, H. (2010). Training and testing of recommender systems on data missing not at random. In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2010)*, pp. 713–722.

Stewart, A., Diaz-Aviles, E., Nejdl, W., Marinho, L. B., Nanopoulos, A., and Schmidt-Thieme, L. (2009). Cross-tagging for personalized open social networking. In: *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia (Hypertext 2009)*, pp. 271–278.

Sun, Y., Han, J., Yan, X., Yu, P. S., and Wu, T. (2011). PathSim: Meta Path-Based Top-K Similarity Search in Heterogeneous Information Networks. In: *2011 Conference on Very Large Database Endowment* 4 (11), pp. 992–1003.

Szomszor, M., Cantador, I., and Alani, H. (2008a). Correlating user profiles from multiple folksonomies. In: *Proceedings of the 19th ACM Conference on Hypertext and Hypermedia (Hypertext 2008)*, pp. 33–42.

Szomszor, M., Alani, H., Cantador, I., O'Hara, K., and Shadbolt, N. (2008b). Semantic Modelling of User Interests Based on Cross-Folksonomy Analysis. In: *Proceedings of the 7th International Semantic Web Conference (ISWC 2008)*. Vol. 5318, pp. 632–648.

Takács, G. and Tikk, D. (2012). Alternating least squares for personalized ranking. In: *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys 2012)*, pp. 83–90.

Tang, J., Yan, J., Ji, L., Zhang, M., Guo, S., Liu, N., Wang, X., and Chen, Z. (2011). Collaborative Users' Brand Preference Mining across Multiple Domains from Implicit Feedbacks. In: *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI 2011)*, pp. 477–482.

Tiroshi, A. and Kuflik, T. (2012). Domain Ranking for Cross Domain Collaborative Filtering. In: *Proceedings of the 20th International Conference on User Modeling, Adaption and Personalization (UMAP 2012)*. Vol. 7379, pp. 328–333.

Tiroshi, A., Berkovsky, S., Kâafar, M. A., Chen, T., and Kuflik, T. (2013). Cross social networks interests predictions based ongraph features. In: *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys 2013)*, pp. 319–322.

Tkalcic, M., Kunaver, M., Košir, A., and Tasic, J. (2011). Addressing the new user problem with a personality based user similarity measure. In: *Proceedings of the 1st International Workshop on Decision Making and Recommendation Acceptance Issues in Recommender Systems*, p. 106.

Vargas, S. and Castells, P. (2011). Rank and relevance in novelty and diversity metrics for recommender systems. In: *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*, pp. 109–116.

Vargas, S. and Castells, P. (2014). Improving sales diversity by recommending users to items. In: *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, pp. 145–152.

Vargas, S., Baltrunas, L., Karatzoglou, A., and Castells, P. (2014). Coverage, redundancy and size-awareness in genre diversity for recommender systems. In: *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, pp. 209–216.

Winoto, P. and Tang, T. (2008). If You Like the Devil Wears Prada the Book, Will You also Enjoy the Devil Wears Prada the Movie? A Study of Cross-Domain Recommendations. In: *New Generation Computing* 26, pp. 209–225.

Xu, S., Bao, S., Fei, B., Su, Z., and Yu, Y. (2008). Exploring folksonomy for personalized search. In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008)*, pp. 155–162.

Xue, G.-R., Lin, C., Yang, Q., Xi, W., Zeng, H.-J., Yu, Y., and Chen, Z. (2005). Scalable collaborative filtering using cluster-based smoothing. In: *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 114–121.

Yao, Y., Tong, H., Yan, G., Xu, F., Zhang, X., Szymanski, B. K., and Lu, J. (2014). Dual-regularized one-class collaborative filtering. In: *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM 2014)*. ACM, pp. 759–768.

Yu, X., Ren, X., Sun, Y., Gu, Q., Sturt, B., Khandelwal, U., Norick, B., and Han, J. (2014). Personalized entity recommendation: a heterogeneous information network approach. In: *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM 2014)*, pp. 283–292.

Zhang, X., Cheng, J., Yuan, T., Niu, B., and Lu, H. (2013). TopRec: domain-specific recommendation through community topic mining in social network. In: *Proceedings of the 22nd International World Wide Web Conference (WWW 2013)*, pp. 1501–1510.

Zhang, Y., Cao, B., and Yeung, D.-Y. (2010). Multi-Domain Collaborative Filtering. In: *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI 2010)*, pp. 725–732.

Zhao, L., Pan, S. J., Xiang, E. W., Zhong, E., Lu, Z., and Yang, Q. (2013). Active Transfer Learning for Cross-System Recommendation. In: *Proceedings of the 27th AAAI Conference on Artificial Intelligence*.

Zhuang, F., Luo, P., Xiong, H., Xiong, Y., He, Q., and Shi, Z. (2010). Cross-Domain Learning from Multiple Sources: A Consensus Regularization Perspective. In: *IEEE Transactions on Knowledge and Data Engineering* 22 (12), pp. 1664–1678.

Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005). Improving recommendation lists through topic diversification. In: *Proceedings of the 14th international conference on World Wide Web (WWW 2005)*, pp. 22–32.