Karlsruher Institut für Technologie

Cross-domain Recommendations based on semantically-enhanced User Web Behavior

Julia Hoxha

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften (Dr. Ing.)
von der Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)
genehmigte Dissertation von
M.Sc. Julia Hoxha

Referent:	Prof. Dr. Rudi Studer, Karlsruher Institut für Technologie
Korreferent:	Prof. Dr. Ricardo Bazea-Yates, Universitat Pompeu Fabra
Tag der mündlichen Prüfung:	28.05.2014

Karlsruhe, 2014

Preface

The enormous quantity of information in the Web vastly outstrips humans' capability to survey it, making it increasingly difficult for users to find the relevant information they seek. An alternative approach to information seeking, besides search engines where an explicit query has to be formulated, is to surf the Web by following appropriate links and interacting with the available Web pages. This is often challenging since people are not always able to determine the links that are most likely to lead to the required information or most relevant Web resources. The task becomes more difficult when we consider that users' interests span over various domains and independent Web sites. Information seeking can be facilitated by recommender systems that guide the users in a personalized manner to relevant resources in the large space of the possible options in the Web. This work investigates how to model people's Web behavior at multiple sites and learn to predict future preferences, in order to generate relevant cross-domain recommendations.

This thesis contributes with novel techniques for building cross-domain recommender systems in an open Web setting. First, we introduce a formal model of user browsing behavior, which is able to capture contextualized knowledge of Web resource through a set of semantic enrichment techniques. The developed artifacts provide a broader context of user behavior, which is used as basis for two cross-domain recommendation approaches.

The first approach comprises a recommendation technique that exploits in novel ways the semantic structures of Web resources in combination with behavior patterns to enable accurate user preference predictions. The contribution also comprises a diversification mechanism to ensure diversity of recommendations from various domains. The second approach addresses the task of inferring user preference relationships to items in a sparse domain by transferring auxiliary knowledge from another domain. The novelty of this work lies in an expressive multi-relational probabilistic model, which is used to facilitate knowledge transfer in a cross-domain collaborative filtering system. We verify the effectiveness of all the presented techniques through a set of different experiments, which are conducted with datasets of real-life user logs from different domains.

Acknowledgements

> *It is good to have an end to journey toward; but it is the journey that matters, in the end.*
>
> Ernest Hemingway

Accomplishing this work has been a long and exciting journey, with ups and downs, small doubts and great joys. At the end of it, I realize that I have learned a lot, met great people, and experienced so much.

I would have not embarked on this very journey, had it not been for my mentor Prof. Rudi Studer who believed in me since the beginning and made me part of his excellent team. I am deeply thankful to him for giving me the freedom to be creative, yet being always present with his advice and mentorship. Prof. Studer is indeed the role model of a wise leader and great scholar.

I would like to thank my colleagues at AIFB and KSRI for creating such a wonderful environment, where it never feels like "work". It is a place full of bright, motivated people who have been part of my life with their advice, feedback, questions, and laughs during each day at the institute. Sincere thanks to my supervisors Stephan Bloehdorn, Achim Rettinger, and Sudhir Agarwal, who each in their own way influenced pieces of my research work.

Before going to Yahoo Labs for my research stay, I would have never thought that I would find these two great researchers and colleagues, Peter Mika and Roi Blanco. I am thankful to them for the incredible support, trust, and the joyful time in Barcelona. I also thank Karlsruhe House of Young Scientists, which financially supported my stay.

I thank my friends for being there, listening to my worries and sharing the joy with me. I am thankful to Anisa and Basil for the many discussions, for the after-lunch coffees, and the laughter that helped to push through the hard work.

ACKNOWLEDGEMENTS

While many people are supporters of my journey, special ones are pillars of my existence. I am immensely grateful to my parents, Ferda and Perparim, who have always believed in me. The only way that I can possibly show them how thankful I am is through my work and accomplishments. I will never stop trying to make them proud.

I had two incredible companions in this journey of mine, my husband Armand and my daughter Emma. I will never be able to find enough words to express how lucky and grateful I am to have Armand in my life. Our love, our partnership in life and work is a continuous source of power that helps me to push my boundaries. All these years would not have been so happy and so easy without Armand's compassion, intelligence, and humor.

My precious Emma, when you grow up and you are able to read these lines yourself, I want you to know that you have inspired me every single day of this journey.

This work is for you and dad!

Contents

Part I

Introduction

CHAPTER 1

Introduction

The enormous quantity of the information in the World Wide Web vastly outstrips humans' capability to survey it, making it increasingly difficult for users to find the relevant information they are seeking. Modern search engines have made the retrieval of information easier and more efficient. Yet, they have an explicit demand for the user to initiate search by formulating a specific search query.

An alternative approach to information seeking is to surf the Web by following appropriate links and interacting with the available Web pages. This is also challenging since people are not always able to determine the links that are most likely to lead to the required information or most relevant Web resources. This task becomes more challenging when we consider the fact that users' interests span over various domains[1] and separate, independent Web sites.

Information seeking can be facilitated by systems that guide the users in a personalized manner to relevant resources in the large space of the possible options in the Web. Such recommender systems [RESNICK and VARIAN 1997] passively observe user behavior when interacting with the Web pages, then use this information to recommend relevant pages from anywhere on the Web. The users are

[1] Domain refers to the set of similar objects with the same characteristics that can be easily differentiated

not required to provide additional input or explicit queries. Behavioral data capturing such Web browsing activities are recorded in collections of click trails, which provide implicit information of user preferences. With the overwhelming growth of the Web, these usage data provide unprecedented opportunities for research on applications of machine learning and reasoning to intelligently guide the users to relevant pages across various Web sites, and, consequently, to support their information-seeking activities.

In this context, the scope of this thesis is to investigate how to model people's Web browsing behavior at multiple websites and learn to predict future preferences based on this model, in order to generate relevant recommendations across different domains. Our goal is to provide techniques for building cross-domain recommender systems in an open Web setting.

The first challenge lies in the fact that users' browsing activities span over several sites, which are usually diverse and highly heterogeneous in their content. Therefore, a crucial step towards better interpretation and analysis of the user behavior across multiple websites is to capture the semantics[2] of the resources that users are accessing. The challenge of information heterogeneity can be tackled by giving meaning to the user browsing activities, relating them to concepts in the domain where they occurred.

Initially, we examine the problem at its root, addressing the task of building a formal model of cross-site user Web browsing behavior based on click trails. Click trails are syntactic representations of requests of the pages and Web resources accessed by the site visitors [SRIVASTAVA et al. 2000]. Due to the primarily syntactical nature of such requests, comprehension of user browsing patterns is difficult. Hence, there is an urge for formalization approaches that leverage the semantics of the usage data in accordance with the domain to which they belong. We present a new model for the formal representation of cross-site user browsing data and a set of novel techniques for the semantic enrichment of these data. The original click trails are transformed into formal semantically-enhanced user behavior models. They provide a broader context of user browsing behavior at various Web

[2] Semantics (from Ancient Greek: sēmantikós) is the study of *meaning*. [LIDDELL and SCOTT 1940]

sites, which can be exploited for intelligent recommendation methods in an open Web setting.

As such, semantic user behavior models are used as basis to build bridges across heterogeneous domains and facilitate information seeking by recommending to human users relevant Web resources to visit next. We exploit the semantically-enhanced models for learning to predict future preferences and, accordingly, suggest to users recommendations that are still relevant, but also come from domains previously unknown to them. We pursue two main approaches to learn user preferential behavior in an open Web setting and, accordingly, generate cross-domain recommendations. Figure 1.1 illustrates the overall framework of cross-domain recommendation techniques and four research questions (denoted with RQ) addressed in this thesis.

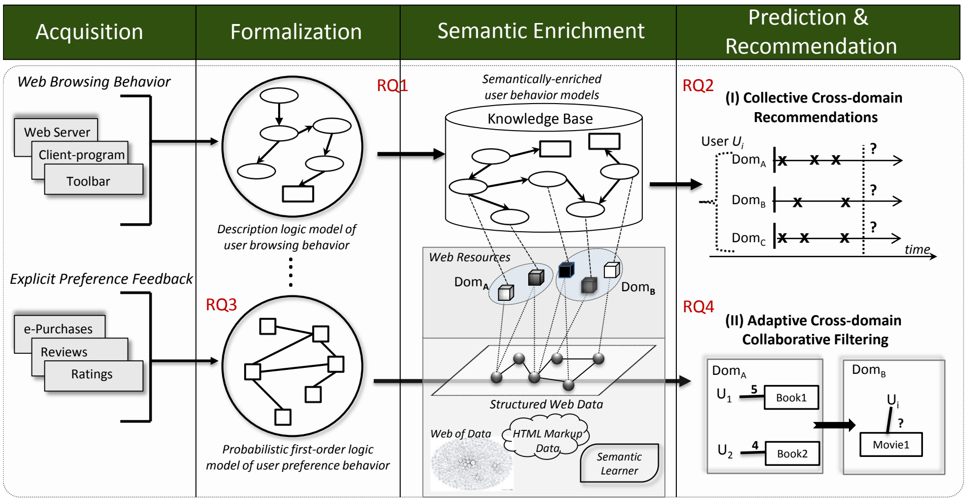


Figure 1.1: Cross-domain recommendation framework

In the first approach, we address the setting when users preferences are implicitly given in their browsing behavior. We capture the preference feedback embedded in their click trails through expressive formal behavior models represented with description logic formalism. Usually, there is little or no overlap among domains and their content is highly heterogeneous. Hence, we provide new techniques for creating connections among resources belonging to different domains by harvesting their semantic representation. These connections, or semantic bridges, are then used as basis in the collective approach we propose for learning to predict *relevant* Web resources and recommend them to users. In order to ensure that the generated list of recommendations is highly *diverse* in terms of the content of resources and domains they belong, we provide a new mechanism for effective diversity enhancement.

In the second approach, we investigate the setting when users have given explicit preferences of Web resources, such as in the form of ratings. User preferential behavior is captured in this case with probabilistic-based logic models, which can be also used for predicting user future preferences in a single domain. Moving to the cross-domain setting, this second approach tackles the recommendation task in an adaptive manner. One of the most successful approaches to build recommender systems based on explicit preference data is collaborative filtering (CF) [PAZZANI 1999], which uses the known preferences of a group of users to make recommendations or predictions of the unknown preferences for other users. Still, a major bottleneck in CF is data sparseness, i.e. restricted user preference data available for making recommendations. In our case, this problem is even more challenging, since the preference data of users across domains not yet explored by them is much more limited. We investigate cross-domain collaborative filtering (CDCF) and present a novel, effective mechanism to alleviate data sparseness of one domain by *transferring knowledge* about user preferences from other domains. The following section presents our main research questions and an outline how each of them will be addressed in this thesis.

1.1 Research Questions

The goal of this thesis is to develop technologies, which enable us to accurately predict user Web browsing preferences and, accordingly, make relevant, cross-domain recommendations in the open Web setting. The requirements entailed by such an open setting impose the *diversity* of recommended Web resources from heterogeneous domains, and *transfer* of knowledge across domains because of user preference sparsity.

We capture this goal in the following main hypothesis substantiated in the thesis:

Main Hypothesis. Accurate cross-domain user recommendations can be generated through predictive techniques, which leverage semantically-enriched models of user Web behavior.

We substantiate the main hypothesis by the following four research questions. Each research question highlights a different facet of the main hypothesis, while focusing on a particular requirement of cross-domain recommendations.

Research Question 1. *How can we model user Web browsing behavior with a formal representation, which semantically leverages the structured descriptions of resources in the Web?*

We address the shortcomings of syntactic representation of user browsing behavior data through a formal and semantic model, which is able to capture contextualized knowledge of Web resources. This problem raises auxiliary questions such as how to discover contextual knowledge from the application domain of the browsed Web resources, and how to achieve semantic enrichment in an open Web setting, without relying on a centralized knowledge base? We investigate this principal research question in Chapter 4.

Research Question 2. *Is it possible to provide a predictive method that captures implicit user preferences and the enriched semantics of Web resources in order to generate accurate recommendations of resources across domains?*

Traditionally, recommender systems have focused on predicting user preferences to items in a single domain. In most of the cases, they are based on historical user data that captures explicit preferences. Yet, the interests of users (i) span across different application domains, and (ii) are not explicitly expressed in the data recorded from their Web browsing behavior.

Through this research question we aim to tackle the challenge of generating recommendations of Web resources across different domains based on a collective approach. Initially, we introduce ways of materializing into measurable metrics the preferences of users implicit in the browsing logs. Furthermore, we exploit ontological information extracted from Web pages to find relations between resources and establish in this way bridges among domains. The final goal is to provide a recommendation approach, which ensures that the generated recommendations are not only relevant to Web visitors, but also highly diverse across domains. We investigate the research question in Chapter 5.

Research Question 3. *Can we build a rich relational model of user behavior that can be used to accurately infer explicit user preferences to make recommendations?*

In this question, we turn our attention to the recommendation task for the case when users express explicit preferences to the objects in the Web, e.g. via ratings. The majority of recommenders addressing this task assume a flat model of data representation, and focus on a single dyadic relationship between objects. We investigate how a richer multi-relational model can be built, so that it allows us to express and reason about many different relations at the same time. We take advantage of the recent progress in statistical relational learning, using a framework that combines logical and probabilistic reasoning. The approach should make it possible to combine many different objects and relations into a comprehensive solution to the recommendation task. We investigate this research question in Chapter 6.

Research Question 4. *When user preference data for resources is very sparse, especially in an open Web setting, how can we transfer user behavior knowledge from one domain to better predict user preferences at a sparse target domain?*

Users more often express only partial preferential feedback by rating limited number of objects, which results in highly sparse user-object relations. To deal with such bottleneck, cross-domain collaborative filtering (CDCF) has been recently studied and shown to help mitigate data sparseness. We investigate how to advance CDCF through an adaptive mechanism, which allows transferring knowledge from a source domain to a sparse, target domain such that we can generate better recommendations. The approach tackles the most challenging case when the users and objects in the two domains are not identical or even overlap. We investigate this research question in the second part of Chapter 6.

1.2 Thesis Contributions

The investigation of the outlined research questions has led to the following four main contributions of the thesis, which also constitute the scientific accomplishment of the author.

Contribution I. *Formal model and semantic enrichment of user Web browsing behavior*

We provide a model for the formal representation of user browsing behavior not restricted to a single Web site, but rather covers multiple sites. We additionally provide three main enrichment techniques: (a) leveraging domain knowledge from Web of Data (fully-fledged domain ontologies), (b) exploiting structured mark-up metadata (microdata, microformat) embedded in HTML, (c) applying machine learning methods to infer semantic types of recommendation resources. This contribution has been presented in previous publications [HOXHA and AGARWAL 2010, HOXHA et al. 2012, HOXHA and AGARWAL 2012] and shortlisted runner-up in the USEWOD Data Challenge [BERENDT et al. 2012]. We present a detailed version of this work in Chapter 4.

Contribution II. *Collective-based technique to generate accurate top-N recommendation of Web resources across domains.*

We provide a technique that applies discriminative learning to make relevant user preference predictions, leveraging the semantic content of resources (captured earlier in the semantic logs). The contribution comprises a diversification mechanism to ensure diversity of recommendations from various domains. The contribution has been presented in a previous publication [HOXHA et al. 2013], while a comprehensive version is under review in a journal [HOXHA et al. 2014a]. A detailed version of this work is presented in Chapter 5.

Contribution III. *Probabilistic first-order model for hybrid recommendations.*

We present an expressive multi-relational model that makes it possible to combine many different objects and relations into a comprehensive solution to the recommendation task. We deploy a hybrid approach for generating recommendations, based on a content/collaborative merging scheme through feature combination. This work has been discussed in a previous publication [HOXHA and RETTINGER 2013]. We present this contribution in the first part of Chapter 6.

Contribution IV. *Adaptive cross-domain collaborative filtering with probabilistic first-order knowledge transfer.*

We extend the expressive relational model of user-object preferences, provided as part of Contribution III, to build a novel effective technique for knowledge transfer from one source domain to another sparse domain. The approach comprises a mechanism for generating accurate recommendations to users in a target domain that is unknown to them. Part of this work has been presented in a previous publication [HOXHA and RETTINGER 2013], and a comprehensive version is under review in a journal [HOXHA et al. 2014b]. We present this contribution in the second part of Chapter 6.

Additional Contributions

Besides these works, the author has also made throughout the course of the PhD studies additional contributions in the field of Semantic Web technologies. These works have been published in various international peer-reviewed venues. A complete list of the publications can be found in Appendix A.1.

1.3 Guide to the Reader

This thesis consists of three parts. The first, introductory part motivates this work, introduces the necessary foundations and state-of-the-art approaches. The second part contains the contributions of this thesis. Initially, we present the approach for the formalization of cross-site user browsing behavior in the open Web. We also introduce a set of techniques for the semantic enrichment of user browsing logs. This second part also contains the contributions of the thesis in building cross-domain recommender systems. The third, concluding part summarizes the work and provides an outlook. In the following, we outline each part and the respective chapters.

Part I: Introduction

Chapter 1 The first chapter motivates this work, discusses its background and outlines its objectives.

Chapter 2 This chapter reviews fundamentals in knowledge representation and necessary machine learning foundations.

Chapter 3 This chapter positions this thesis with respect to related works. It gives an overview of the state-of-the-art techniques in semantic recommender systems, cross-domain recommender systems, and hybrid recommender systems, at the intersection of which our work resides.

Part II: Cross-domain Recommendation Models based on User Web Behavior

Chapter 4 In this chapter, we present a formal model of user Web browsing behavior at multiple sites based on the click trails (logs), tackling the problem of information heterogeneity by using semantics. We introduce a novel two-staged approach for the semantic enrichment of usage logs with domain knowledge, bringing together Semantic Web technologies and machine learning techniques.

In the first stage, we present two methods for extracting domain-level structured objects as semantic resources contained in the pages that the users have visited. The second stage consists of a supervised learning approach to find missing content types of the resources. The semantic formalization of user browsing behavior lays the basis for effective techniques of querying expressive usage patterns.

Hence, as an extension of our formalization approach, we append in Appendix B.1 a querying formalism based on a temporalized description logic, which combines temporal logic with ontological reasoning capabilities. We also present a query answering mechanism that enables the discovery of expressive usage behavior patterns through the formulation of semantic and temporal-based constraints.

Chapter 5 This chapter addresses the challenging problem of making predictions of user browsing preferences in the open Web setting. Our objective is to generate cross-domain recommendations of Web resources by exploiting the semantic logs derived in Chapter 4. These are the observations of user browsing behavior enriched with the semantic structures extracted from Web pages. This information is beneficial to learn commonalities across different domains in terms of the semantic similarity of the Web resources. The information is used as basis to predict future relevant resources to be suggested to the users in their cross-domain Web browsing activity.

In this context, recommendations need to be both relevant and diverse to help users explore novel topics and find information sources previously unknown to them. Our work tackles the problem of recommending Web resources from multiple domains, while at the same time balancing the relevance and diversity of recommendations.

Chapter 6 In this chapter, we exploit rich semantic models of user behavior, focusing on explicit user preferences about objects across various domains. We address the task of inferring user preference relationships to items in one domain by transferring auxiliary knowledge from another domain. In the context of open Web, where there is often much more relational information available than a single user-item relationship, we need inference techniques that are able to capture multi-relational information.

We present a richer theoretical model for making recommendations, which allows us to reason about many different relations at the same time. The semantic data is captured in this relational model to extend the attributes of each item. The novelty of this work lies in using an expressive multi-relational model to ease knowledge transfer in a cross-domain collaborative recommender system setting.

Part III: Conclusions and Outlook

Chapter 7 This chapter summarizes the contributions of this thesis and draws conclusions. It also points out the limitations of this work and gives insights on future work.

CHAPTER 2

Foundations

In this chapter, we introduce the foundations for the work presented in the thesis. A core requirement for our work is that the knowledge capturing user Web behavior is represented in a machine-understandable form. We thus recapitulate in the first part of the chapter (Section 2.1) formalisms for knowledge representation, introducing first-order logic, description logics, and ontology languages in the Semantic Web. The second part of the chapter focuses on the other theoretical background needed for our work, namely machine learning (Section 2.2). We start with the introduction of traditional non-relational learning methods that use feature-based representations. We focus particularly on Support Vector Machines as a representative technique that is also applied in our work. We then continue with techniques of Statistical Relational Learning, which are based on multi-relational representations of data. We particularly focus on Markov Logic Networks used in our work.

2.1 Knowledge Representation Formalisms

Knowledge representation, as one of the central concepts in Artificial Intelligence, is devoted to make the information about the world explicit, in a form that enables computer systems to understand it and use it for problem solving [NEWELL 1982]. The technologies for such representation of knowledge build upon formal languages that are unambiguous, logically adequate representation of natural language, and allow to infer new, implicit knowledge. The later is referred to as reasoning, which is a key inference capability to derive new statements from the existing statements in the knowledge. Besides producing new knowledge, reasoning also enables consistency checking of existing knowledge.

Technologies of knowledge representation are utilized to describe particular domain models, which capture the *domain knowledge* as a formal representation of the knowledge of the experts in that domain. The set of statements that hold according to the particular *conceptualization* of a domain are referred to as *ontology*, whereby conceptualization denotes an abstract view of the world [GRUBER 1995, STUDER et al. 1998]. Minimally, an ontology makes explicit assertions about the entities of interest in a given domain through logical statements, also referred to as *axioms*. Expressing these axioms is a crucial task in the engineering of ontologies, which requires the use of logic formalisms. Formal languages that are well-understood and often used for this task include **first-order logic** (FOL) and its decidable subset **Description Logics** (DL), which have become very prominent in the Web. In the following, we give an introduction to this class of logics, focusing on the key properties of DL applied in our work.

2.1.1 First-order Logic

First-order logic is a formalism that enables the representation of compactly complex relational structures [GENESERETH and NILSSON 1987]. A *first-order knowledge base* (KB) is a set of formulae in first-order logic, which are composed of four types of symbols: predicates, functions, variables, and constants.

Constants represent objects in a domain of interest (e.g. people: $Sara$, Bob, etc.). Variable symbols range over the objects. Functions represent a mapping from tuples of objects to objects (e.g. AuthorOf). Predicate symbols represent relations betwen objects (e.g. hasRating) or attributes of objects (hasAge). Variables and constants may be typed, in which case variables only range over objects of the given type.

Syntax

In general, a formal language is a recursively defined set of strings on a fixed alphabet. In FOL, the *signature* $\Sigma = P \cup C$ is the union of a set of *predicates* P and a set of *constants* C, such that $P \cap C = \emptyset$.

The arity refers to the number of argument places in a predicate. For every integer $n \geq 0$, we have a set P^n of n-place predicates and a set F^n of n-place functions. We call predicates with an arity of $n > 1$ *relations*.

A *term* is any expression representing an object in the domain. It can be a constant, a variable, or a function applied to a tuple of terms. The set V of individual *variables* is disjoint to the signature. Constants and variables are FOL *terms* that serve as a building block of FOL formulae.

Atomic and Compound Formulae. An *atomic formula* or *atom* is a predicate symbol applied to a list of terms (e.g. $\texttt{hasRated}(Anna, book_1, 5)$. An atomic formula is an expression of the form $p(t_1, \ldots, t_n)$, where $p \in P$ is a predicate of arity n, and each $t_i \in C \cup V$ is a term. Additional atomic formulae are $\top$ and $\bot$, which represent the Boolean values true and false, respectively.

Compound formulae are recursively constructed from atomic formulas using logical connectives (symbolized $\wedge, \vee, \neg, \Rightarrow$), and quantifiers: the universal quantifier ($\forall$), or the existential quantifier ($\exists$). Parentheses may be used to enforce precedence. Thereby, if ϕ and ψ are formulae and x is a variable, then the following $\neg\phi$, $\phi \wedge \psi$, $\phi \vee \psi$, $\phi \rightarrow \psi$, $\exists x.\phi$, $\forall x.\phi$ are also formulae.

A *positive literal* is an atomic formula; a *negative literal* is a negated atomic formula. The formulas in a KB are implicitly conjoined, and we can view a KB as a large single formula.

Ground Terms. A term is ground when it contains no variables, but all its arguments are constants. A ground atom or ground predicate is an atomic formula all of whose arguments are ground terms.

Free and bound variables. In a formula, a variable can occur as free or bound. A variable is free in a formula if it occurs outside of the scope of a quantifier $(\exists, \forall)$, otherwise it is bound. Variables of atomic formulae are free.

Interpretation

The semantic of first-order logic is defined by the interpretation $(\Delta^{\mathcal{I}}, \mathcal{I})$ composed of the domain of discourse Δ and intepretation function $\mathcal{I}$. The domain Δ denotes an abstract set of individuals that represent the universe in which the symbols of Σ are interpreted. The interpretation function $\mathcal{I}$ assigns:
(i) to every constant $c \in C$ an element $\mathcal{I}(c) \in \Delta$ in the domain, and
(ii) to every predicate $p \in P$ with arity n a relation $\mathcal{I}(p) \subseteq \Delta^n$.
An interpretation assigns to every formula ϕ a truth value $\mathcal{I}(\phi)$, which can be inductively defined in the following way:

$$\begin{aligned}
\mathcal{I}(\top) &= \text{true} && \\
\mathcal{I}(\bot) &= \text{false} && \\
\mathcal{I}(p(t_1, \ldots, t_n)) &= \text{true}, && \text{iff} \quad (\mathcal{I}(t_1), \ldots, \mathcal{I}(t_n)) \in \mathcal{I}(p) \\
\mathcal{I}(\phi \wedge \psi) &= \text{true}, && \text{iff} \quad \mathcal{I}(\phi) = \text{true and } \mathcal{I}(\psi) = \text{true} \\
\mathcal{I}(\phi \vee \psi) &= \text{true}, && \text{iff} \quad \mathcal{I}(\phi) = \text{true or } \mathcal{I}(\psi) = \text{true} \\
\mathcal{I}(\phi \Rightarrow \psi) &= \text{false}, && \text{iff} \quad \mathcal{I}(\phi) = \text{true or } \mathcal{I}(\psi) = \text{false} \\
\mathcal{I}(\neg\phi) &= \text{true}, && \text{iff} \quad \mathcal{I}(\phi) = \text{false} \\
\mathcal{I}(\forall v.\phi) &= \text{true}, && \text{iff} \quad \mathcal{I}(\phi_{v \leftarrow c}) = \text{true for all } c \in C \\
\mathcal{I}(\exists v.\phi) &= \text{true}, && \text{iff} \quad \text{there exists } c \in C \text{ such that } \mathcal{I}(\phi_{v \leftarrow c}) = \text{true}
\end{aligned}$$

A set of formulae compose a *theory* T. When all possible ground atoms of the theory have been assigned truth values along an interpretation $(\Delta^{\mathcal{I}}, \mathcal{I})$, they form a *possible world.* A formula is *satisfiable* iff there exists at least one world in which it is true.
The central problem in logic is the *inference* problem, which consists in determining whether a theory T entails a formula ϕ, denoted by $T \models \phi$. This consists in checking if ϕ is true in all those worlds where the theory is satisfied, i.e. each of the formulae in T is true under that interpretation.

Clausal Form. In order to automate the inference process, it is convenient to convert the formulae to clausal form, also known as conjunctive normal form (CNF). A formula in this form is referred to as a *clause*, which is a disjunction of literals. Hence, the theory consists of the conjuction of clauses. Every theory can be converted to clausal form by following a mechanical sequence of steps. In our work, we express the formulae in their clausal form. Table 2.1 shows a set of simple formulae and their representation in clausal form.

First-Order Logic	Clausal Form
"People like the books of a particular author"	
$\forall u \forall b_1 \forall b_2\ \texttt{Au}(b_1, b_2) \Rightarrow (\texttt{li}(u, b_1) \Leftrightarrow \texttt{li}(u, b_2))$	$\neg\texttt{Au}(b_1, b_2) \vee \texttt{li}(u, b_1) \vee \neg\texttt{li}(u, b_2)$
	$\neg\texttt{Au}(b_1, b_2) \vee \neg\texttt{li}(u, b_1) \vee \texttt{li}(u, b_2)$
"People of similar age like the same books"	
$\forall u_1 \forall u_2 \forall b\ \texttt{Ag}(u_1, u_2) \Rightarrow (\texttt{li}(u_1, b) \Leftrightarrow \texttt{li}(u_2, b))$	$\neg\texttt{Ag}(u_1, u_2) \vee \texttt{li}(u_1, b) \vee \neg\texttt{li}(u_2, b)$
	$\neg\texttt{Ag}(u_1, u_2) \vee \neg\texttt{li}(u_1, b) \vee \texttt{li}(u_2, b)$

Table 2.1: Example of first-order formulae in clausal form. $\texttt{Au}()$ is short for $\texttt{shareAuthor}()$, $\texttt{Ag}()$ for $\texttt{shareAge}()$, and $\texttt{li}()$ for $\texttt{likes}()$.

Completeness and Decidability. The inference problem in first-order logic is a semi-decidable decision problem. For a decision problem, it is formally required to know if it is possible to construct a single algorithm that always leads to a correct yes-or-no answer. A problem is called semi-decidable, or partially decidable if there exists an algorithm that stops eventually when the answer is yes, but may run forever if the answer is no. For FOL, Gödel's completeness theorem establishes that there exists an effective, sound, and complete deductive system that eventually captures the logical consequence relation by finite provability.

Yet, since inference is not fully decidable, theories are often constructed using decidable, restricted subsets of first-order logic. The subsets most commonly applied to the Semantic Web are description logics. In the next section, we give a general introduction to description logic, then present the expressive $\mathcal{SHOIN}(\mathbf{D})$ fragment of description logics, which we have applied in our work.

2.1.2 Description Logics

Description logics (DL) are a decidable fragment of first-order logic, whose development has been driven by the need to increase expressivity of knowledge representation formalisms, while still maintaining decidability [Baader and Nutt 2003]. Description logics provide formal semantics that allow to specify precise meaning of the concepts in a domain. Earlier results on DL have been utilized for ontological modeling in the Semantic Web, forming also the basis of the Web Ontology Language (OWL) [Bechhofer et al. 2004].

An extensive introduction of DL is presented in the work of Baader *et al.* [Baader and Nutt 2003]. We restrict to an overview of the key features of DL, presenting the most basic fragment called $\mathcal{ALC}$, and the description logic $\mathcal{SHOIN}(\mathbf{D})$ that is used in our work. $\mathcal{SHOIN}(\mathbf{D})$ is the logic formalism underpinning OWL.

There are three main building blocks in DL: individuals, concepts, and roles. The symbols referred to as constants in first-order logic are referred to as *individuals* in DL. *Concepts* in a DL correspond to unary predicates denoting sets of individuals, whereas *roles* correspond to binary predicates, i.e. relations between individuals.

The state of a domain is represented by a DL *ontology*, which is a set of DL statements (axioms) that are true in that state. There are two types of axioms in DL: terminological axioms and assertional axioms. The set of all terminological axioms form the TBox (short for *terminology box*), whereas the set of assertional axioms form the ABox (short for *assertion box*, which asserts statements about a given terminology). As such, concepts and roles are separated from individuals, but TBox and ABox altogether compose the *knowledge base* (KB).

We adhere to the common notation in which capital letters A, B denote atomic concepts, C, D denote complex descriptions of concepts, R, S denote atomic roles, and c, d are individuals. Complex concepts and complex roles can be built by using concept constructors, which include the universal concept ($\top$), bottom concept ($\bot$), atomic negation ($\neg$), intersection ($\sqcap$), value restrictions ($\forall R.C$), and existential quantification ($\exists R.C$).The supported concept and role constructors in a specific DL determine its expressive power [Grosof et al. 2003].

Description Logic $\mathcal{ALC}$

A simple description logic with a basic set of language constructs and low expressivity is the Attributive Language with Complements $\mathcal{ALC}$ [SCHMIDT-SCHAUSS and SMOLKA 1991]. Table 2.2 illustrates the syntax and semantics of the concept constructors in $\mathcal{ALC}$ given the interpretation $(\Delta^{\mathcal{I}}, \mathcal{I})$. We also give in Table 2.2 a short description for each constructor.

Constructor	Syntax	Semantics	Description
Universal Concept	$\top$	$\Delta^{\mathcal{I}}$	set of all individuals
Bottom Concept	$\bot$	$\emptyset$	empty set
Intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	set of individuals that are both in $C^{\mathcal{I}}$ and $D^{\mathcal{I}}$
Union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	set of individuals that are in $C^{\mathcal{I}}$ or $D^{\mathcal{I}}$
Complement	$\neg C$	$\Delta^{\mathcal{I}} - C^{\mathcal{I}}$	set of individuals that are not in $C^{\mathcal{I}}$
Existential Restriction	$\exists R.C$	$\{c \mid \exists (c, d) \in R^{\mathcal{I}} \wedge d \in C^{\mathcal{I}}\}$	set of individuals that are related via the role $R^{\mathcal{I}}$ to an individual in $C^{\mathcal{I}}$
Universal Restriction	$\forall R.C$	$\{c \mid \forall (c, d) \in R^{\mathcal{I}} \Rightarrow d \in C^{\mathcal{I}}\}$	individuals that are related via the role $R^{\mathcal{I}}$ only to individuals in $C^{\mathcal{I}}$

Table 2.2: Constructs in $\mathcal{ALC}$ description logic

A DL knowledge base consists of a set of assertional axioms and a set of terminological axioms. In Table 2.3, we summarize the syntax and semantics of the axioms that can be expressed in $\mathcal{ALC}$. In the table, we also give a short informal description for each axiom. Subsumption in DL is defined as an inclusion axiom $A \sqsubseteq B$, read as "A is subsumed by B" (or "B subsumes A"), which is also interpreted as B is a subclass of A.

An important feature that can be integrated in description logics is the inclusion of built-in predicates modeling concrete properties with values from a fixed domain, e.g. integers or strings. As such, DLs may be extended with *concrete domains* that

TBox Axiom	**Syntax**	**Semantics**	**Description**
Concept Equivalence	$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$	C is equivalent to D, i.e. every individual in $C^{\mathcal{I}}$ is also an individual in $D^{\mathcal{I}}$, and vice versa
Role Equivalence	$R \equiv S$	$R^{\mathcal{I}} = S^{\mathcal{I}}$	R is equivalent to S, i.e. every pair of individuals in the set $R^{\mathcal{I}}$ also belongs to the set $S^{\mathcal{I}}$, and vice versa
Concept Subsumption	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$	C is a subconcept of D, i.e. every individual in the set $C^{\mathcal{I}}$ is also an individual in $D^{\mathcal{I}}$
Role Subsumption	$R \sqsubseteq S$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$	R is a subrole of S, i.e. every pair of individuals in $R^{\mathcal{I}}$ belongs also to the set $S^{\mathcal{I}}$
ABox Axiom	**Syntax**	**Semantics**	**Description**
Concept Assertion	$C(c)$	$c^{\mathcal{I}} \in C^{\mathcal{I}}$	c is an instance of concept C, i.e. $c^{\mathcal{I}}$ is in the set $C^{\mathcal{I}}$
Role Assertion	$R(c, d)$	$(c^{\mathcal{I}}, d^{\mathcal{I}}) \in R^{\mathcal{I}}$	c is related via R to d
Individual Equivalence	$c \equiv d$	$c^{\mathcal{I}} = d^{\mathcal{I}}$	$c^{\mathcal{I}}$ and $d^{\mathcal{I}}$ are the same individual
Individual Inequivalence	$c \neq d$	$c^{\mathcal{I}} \neq d^{\mathcal{I}}$	$c^{\mathcal{I}}$ and $d^{\mathcal{I}}$ are different individuals

Table 2.3: Axioms in $\mathcal{ALC}$ description logic

can be used to build complex concepts, for example the following axiom describes adults as humans whose age is at least 18: $\mathsf{Adult} \equiv \mathsf{Human} \sqcap \exists age. \geq_{18}$.

When this feature is integrated in a logic, its name is extended accordingly with the symbol D. Hence, the description logic ALCD integrates concrete domains D in the basic ALC. The feature of concrete domains is useful for the practical applications of DL.

Description Logic $\mathcal{SHOIN}(\mathbf{D})$

A more expressive description logic, which is also the formalism underlying the ontology languages in the Web, is $\mathcal{SHOIN}(\mathbf{D})$ description logic. The naming also reflects its expressive power. Specifically, $\mathcal{SHOIN}(\mathbf{D})$ extends $\mathcal{ALC}$ with features of role transitivity, inverse roles, role hierarchy, nominals, cardinality restrictions, and datatypes.

Constructor	Syntax	Semantics
Qualified (atleast) Restriction	$\geq nR$	$\{c \mid \#\{d \mid (c,d) \in R^{\mathcal{I}}\} \geq n\}$
Qualified (atmost) Restriction	$\leq nR$	$\{c \mid \#\{d \mid (c,d) \in R^{\mathcal{I}}\} \leq n\}$
Enumeration	$\{c_1, \ldots, c_n\}$	$\{c_1^{\mathcal{I}}, \ldots, c_n^{\mathcal{I}}\}$
Inverse Role	R^-	$\{(d,c) \mid (c,d) \in R^{\mathcal{I}}\}$
Role Transitivity	$Trans(R)$	$R^{\mathcal{I}} = (R^{\mathcal{I}})^+$

Table 2.4: Constructs in $\mathcal{SHOIN}(\mathbf{D})$ description logic in addition to $\mathcal{ALC}$ constructs

The constructor for $\mathcal{SHOIN}(\mathbf{D})$ are those used in $\mathcal{ALC}$ (Table 2.2) extended with additional constructors that we present in Table 2.4.

2.1.3 Ontology Languages in the Semantic Web

Formal representation of knowledge using ontologies has played a crucial role in the prominence of Semantic Web, which aims at making the information on the Web understandable for machines. The Web Ontology Language (OWL) is a W3C standard for the ontology formalization [BECHHOFER et al. 2004]. OWL consists of a family of languages that are based on description logics. There are three main profiles of OWL: OWL-Lite, OWL-DL, and OWL-Full. Each of them offers a different expressive power based on the particular description logic used as underlying formalism. OWL-Lite has a low degree of expressivity, for example it does not feature quantification. On the other side, OWL-Full is very expressive, but is known to be undecidable and lacks practicality for available reasoners. OWL-DL is expressive, yet is "carefully crafted to remain decidable" for practical implementations [GROSOF et al. 2003].

The formalism underlying OWL-DL is the description logic $\mathcal{SHOIN}(\mathbf{D})$, which as mentioned earlier is used in our work for user behavior formalization. OWL-DL is compatible with a number of inference engines, such as Pellet [SIRIN et al. 2007], which we also use for consistency checking of the engineered ontology. The increasing interest in Semantic Web and description logic formalisms has triggered the development of tools and utilites, such as for example ontology editors Protégé [NOY et al. 2001] and WebProtégé [TUDORACHE et al. 2013], particularly useful in our work for creating ontologies and checking consistency with reasoners.

A new version of OWL, called OWL 2 [OWL WORKING GROUP 2009], is currently available. It keeps an overall structure similar to the previous version, but offers an increased expressivity.

Serialization Formats

In order to assure interoperability and facilitate information sharing in the Web, OWL can be serialized in different formats, such as XML/RDF, OWL/XML, OWL Functionaly Syntax, and Manchester OWL Syntax. In particular, it was designed to be compatible with the eXtensible Markup Language (XML) by extending the Resource Description Format (RDF), leading to XML/RDF which is also used in our work. Syntactically, an OWL ontology is an RDF document in a well-formed XML syntax.

The RDF W3C recommendation [MANOLA and MILLER 2004] is a graph-based data model with labeled nodes and directed, labeled edges. It offers a lightweight representation of data entities and relationships in the Web [PAN 2009]. The fundamental unit of RDF is a statement that corresponds to an edge in the graph. An RDF graph is formed by a set of statements. An RDF statement consists of three components: subject, predicate, object. The subject is the source of the edge, modeled by the node and represents a resource. A *resource* is any entity uniquely identifiable with a Uniform Resource Identifier (URI), which in most of the cases is a Uniform Resource Locator (URL) used to identify the address of a Web page. The object of an RDF statement is the target of the edge, also modeled as a node. The object can be a resource identifiable by a URI, or alternatively a literal value e.g. number or string. The predicate of an RDF statement denotes the relationship between the subject and an object, represented by the label of the edge and also identified by a URI.

An element or attribute name in RDF can be given as a qualified name of the form prefix:local_name, which uses XML namespaces to provide shorthand references for URIs. XML namespace URIs in RDF are useful to distinguish between properties with the same name, especially when different schemas are referenced in the same RDf graph. A comprehensive introduction of the RDF language is given in the W3C specification [MANOLA and MILLER 2004], whereas a more detailed overview of the RDF model theory and semantics is presented in [PAN 2009, HAYES 2004].

2.2 Machine Learning Foundations

Machine Learning is a broad field, which is concerned with the construction and study of systems that can learn from data and improve their performance with experience [MITCHELL 1997]. Many different kinds of problems can be solved using learning systems, in particular prediction and recommendation that are the focus of this thesis. We use the following well-known definition of a learning system [MITCHELL 1997]:

Definition 1. (Learning System) *A computer program is said to* learn *from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P improves with experience E.*

Learning systems are based on observations or experiences that improve the system performance. These experiences are made available to the machine learning algorithm as *input data*. Problems addressed in ML are generally divided into *unsupervised* and *supervised* learning problems. Unsupervised learning problems aim at discovering in the data patterns that are not known a priori. In our work, we address supervised learning problems that consist in solving the so-called *prediction tasks*. In these tasks, each observation is composed of two parts: a *target value* y_i and the *input instances* x_i defined by input values x_{ij}. The learning problem consists in building a function f, so that $f(x_i) = y_i$ for all observations (in the optimal case).

Each learning problem is formulated through the following components:

- Suitable data representation, i.e. a suitable data model and a suitable description of each element from the input data in the data model
- Suitable class of hypotheses $\mathcal{H}$ (the *model class*).
- Method for choosing a specific model function $f \in \mathcal{H}$ by adjusting the free parameters of functions within $\mathcal{H}$.

Traditional approaches to the problem derive from classical work in statistical pattern recognition. This research has mainly focused on attribute-value or propositional learning algorithms, which generally assume that input data is represented as points in a high-dimensional space. The systems based on such approaches offer elegant solutions to prediction problems and perform efficiently. On the other side, they hide the rich logical structure of the data and potential relations lying underneath.

In the recent years, there has been an increasing interest on machine learning approaches that consider the structured representation or the relational model of the data, which could be further useful for complex problem solving. The strongest motivation for using a relational model is the ability to model dependencies between related objects in the data. As such, the information about one object is used to reach conclusions about other, related objects. This research area is referred to as Statistical Relational Learning (SRL) [GETOOR and TASKAR 2007].

In our work, we have investigated representative approaches from both families of non-relational and relational learning for the task of cross-domain recommendation. As needed preliminaries of our work, we present in this chapter the fundamental learning algorithms that we have used from each family. We start with the traditional non-relational approach with feature-based representations, focusing on SVM as a representative technique, which is also applied in our work. We continue with an SRL technique based on multi-relational representations, more specifically the Markov Logic Networks.

2.2.1 Support Vector Machines

Propositional learning is based on classical statistical approaches that aim at generalizing a hypothesis from observations on a few statistical units, where each unit is an entity in the variables or features of interest. The set of observations is also referred to as training data. When investigating a population of statistical units, the quantities of interest are the features that are extracted from the units themselves. Traditionally, ML problems deal with the representation of each unit with M attributes as a vector of features in the form $\mathbf{x} =< x_1, ..., x_n >$. This vector captures the attributes of the unit, which are in most cases binary, real valued, or discrete.

In the common discriminative setting, the vector of features is accompanied by a single output y, referred to as label or class. The format of feature vector with an output label is popular because of its intuitive representation and the simplicity it offers in supporting the i.i.d (for independent and identically distributed) assumption of variables in the learning task. The elements in the vector x are referred to as the independent variables, whereas y is the dependent variable relying on $\mathbf{x}$. Given this setting, the task at hand is to find a function f, so that $f(x_i) = y_i$, which is able to discriminate, or classify in a correct way each input vector assigning it to an output class.

The most basic classification model is the family of linear classifiers, which aim at finding an hyperplane to separate the instances in the input space according to the position of each instance w.r.t. the hyperplane. Thus, given a set of input feature vectors $\mathbf{X}$ and two classes y_1 and y_2 (namely possible values of the output variable), the problem is to find a hyperplane described by its normal vector $\mathbf{w}$ and the bias term b such that

$$f(x) =< w, x > +b \begin{cases} \geq 0 & \text{if } x \in y_1 \\ < 0 & \text{if } x \in y_2 \end{cases}$$

The task consists in finding a hyperplane that is able to separate the data: objects belonging to class y_1 are on one side of the separating hyperplane, objects of class y_2 on the other. An alternative way to learn linear classifiers is provided by Support Vector Machines (SVMs). In fact, SVMs extend linear models in order to classify also data that are not linearly separable.

Support Vector Machines (SVMs) are a set of classification techniques initially introduced in [Boser et al. 1992] and refined by [Vapnik 1995]. They are widely popular, mainly because of their ability to handle high-dimensional data and be robust to noise in the data. They are also shown to be very accurate in many classification problems of various application areas. SVMs do not find *any* separating hyperplane, rather the one with the maximum margin, i.e. maximum *minimum distance* between a training instance and the separating hyperplane.

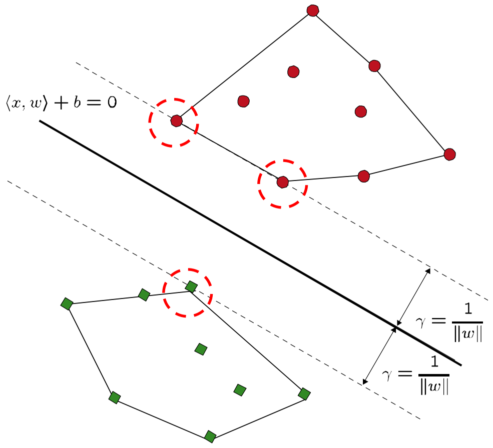

$\langle x, w \rangle + b = 0$
$\gamma = \frac{1}{\|w\|}$
$\gamma = \frac{1}{\|w\|}$

Figure 2.1: SVM separating hyperplane maximizing the margin is denoted by a solid line. The dotted lines are the ones that lines that fix the margin, whereas the points on the margin are the support vectors.

The maximum margin hyperplane provides the greatest separation between classes. Among all possible hyperplanes that separate the classes, the maximum margin hyperplane is the one that is as far away as possible from the two convex hulls, where each hull is formed by connecting the instances of a class. This situation is illustrated in Figure 2.1.

Definition 2. (Functional Margin) *The* functional margin γ *of a hyperplane* (w, b) *with respect to a data point* (x_i, y_i) *is defined as the quantity*

$$\gamma_i = y_i < w, x_i > +b$$

The geometric margin *is obtained by rescaling w and b. It then represents the Euclidean distance of x_i from the hyperplane:*

$$\gamma_i = y_i < \frac{1}{||w||} w, x_i > + \frac{1}{||w||} b$$

The rationale for selecting the hyperplane with the maximal margin is that classifiers with a larger margin tend to provide better generalization.

Definition 3. (Hard-margin SVM) *Hard-margin SVM are linear classifiers based on the maximum margin hyperplane, which in the case of linear separable data can be found as solution of the constrained optimization problem:*

$$\begin{aligned} &\min_{w} \frac{||w||^2}{2} \\ \textit{subject to} \quad & y_i(< w, x_i > + b) \geq 1, \ i = 1, \ldots, n \end{aligned}$$

The objective function is quadratic and the constraints are linear in the parameters w and b. As such, the optimization problem is known to be convex and it can be solved using Lagrangian multipliers. After reformulation, this leads to the dual optimization problem:

Definition 4. (Hard-margin SVM - Dual Optimization Problem)

$$\begin{aligned} &\max_{\alpha} \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j < x_i, x_j > \\ \textit{subject to} \quad & \sum_{i=1}^{l} y_i \alpha_i = 0, 0 \leq \alpha_i, \forall i \end{aligned}$$

Most Lagrange multipliers α_i in the solution are equal to zero. If $\alpha_i \neq 0$, the distance of training instance x_i to the separating hyperplane equals the geometric margin. The training instances that safisfy such condition are called *support vectors*. This is also illustrated in Figure 2.1. There is at least one support vector for each class, and often there are more. A set of support vectors can uniquely define the maximum margin hyperplane for the learning problem. All other training instances are irrelevant; they can be removed without changing the position and orientation of the hyperplane.

2.2.2 Markov Logic Networks

While the propositional learning approaches assume a rather "flat" or vectorial data representation for each object, focusing on a single dyadic relationship between the objects, the field of statistical relational learning investigates structured, relational models. We take advantage of the progress in SRL, which provides rich representations and efficient inference and learning algorithms for non-i.i.d. data. In particular, we use Markov logic, which combines first-order logic and Markov random fields [Sarwar et al. 2000], resulting in the refined probabilistic models of Markov Logic Networks (MLNs), which have emerged as a powerful and popular framework combining logical and probabilistic reasoning.
A key advantage of MLNs is that they allow to express semantically-rich formulae to capture a variety of dependencies between entities in a seamless fashion. MLNs can be used for reasoning about an entity using the entire rich structure of knowledge encoded by the relational representation.

Markov Networks

A Markov network (also known as Markov random field) is a model for the joint distribution of a set of variables $X = (X_1, X_2, ..., X_n)$. It is composed of an undirected graph G and a set of potential functions ϕ_k. The graph contains a node for each variable, and the model has a potential function for each clique in the graph. The joint distribution is:

$$P(X = x) = \frac{1}{Z} \prod_{x} \phi_k(x_{\{k\}}), \tag{2.1}$$

where $x_{\{k\}}$ is the state of the variables that appear in the k-th clique, and Z is the partition function $Z = \sum_x \prod_x \phi_k(x_{\{k\}})$.
Provided that $\forall x, P(X = x) > 0$, Markov networks are also conveniently represented as log-linear models:

$$P(X = x) = \frac{1}{Z} \text{exp}(\sum_{x} w_i f_i(x)) \tag{2.2}$$

Each clique potential is replaced by an exponentiated weight-ed sum of features of the state. There is one feature, which may be any real-valued function of the state, corresponding to each possible state $x_{\{k\}}$ of each clique, with its weight being log. While this representation is exponential in the size of the cliques, one can specify much smaller number of features, like logical functions of the state of the clique. This leads to more compact representations, especially in the presence of large cliques.

In such probabilistic models, the goal is to find the most likely state of a set of query (unobserved) variables given the state of a set of evidence variables (this is inference), and to compute the conditional probabilities of unobserved variables (conditional inference). These problems are #P-complete and solutions widely used resort to approximations like Markov chain Monte Carlo (MCMC) and Gibbs sampling.

Markov Logic

In first-order logic, the formulae are hard constraints on the set of possible worlds: if a world violates even one formula, it has zero probability. In many applications, there is a need to soften these constraints: when a world violates one formula in the KB it is less probable, but not impossible. This is also the setting that Markov Logic Networks provide. Each formula has an associated weight that reflects how strong is this constraint. We use the following definition of Markov Logic Network [RICHARDSON and DOMINGOS 2006]:

Definition 5. (Markov Logic Network) *A Markov logic network (MLN) L is a set of pairs (F_i, w_i), where F_i is a formula in first-order logic and w_i is a real number. Given a set of constants C, it defines a Markov Network $M_{L,C}$ as follows:*

- *$M_{L,C}$ contains one binary node for each possible grounding of each predicate appearing in L. The value of the node is 1 if the ground predicate is true, and 0 otherwise.*
- *$M_{L,C}$ contains one feature for each possible grounding of each formula $F_i \in L$. The value of this feature is 1 if the ground formula is true, and 0 otherwise. The weight of the feature is the w_i associated with F_i in L.*

Based on Definition 5, the graphical structure of $M_{L,C}$ is obtained as follows: there is an edge between two nodes in $M_{L,C}$, iff the corresponding ground predicates appear together in at least one grounding of one formula in L. An MLN serves as a template for constructing Markov networks.

1.1	$\forall u_1, u_2, b\ shareAge(u_1, u_2) \Rightarrow \big(likes(u_1, b) \Leftrightarrow likes(u_2, b)\big)$

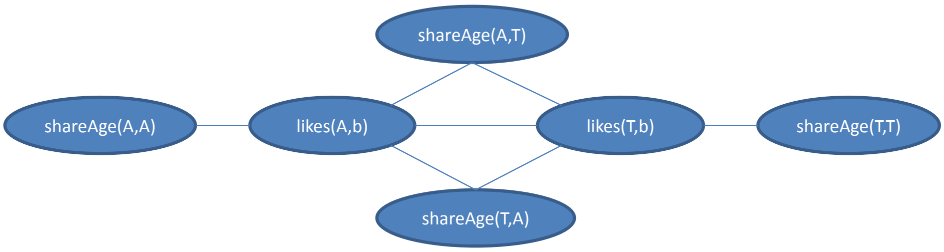


Figure 2.2: Ground Markov Network obtained by applying the formula the constants Anna (A), Tom (T), and book (b)

Figure 2.2 illustrates the graph of the Markov network defined by the given formula in markov logic, and the constants Anna (A), Tom (T), and book (b). Each node of the graph is a ground atom, for example *shareAge(A,T)* or *likes(A,b)*. There is an edge between every two atoms that appear together in some grounding of the formula. The constructed $M_{L,C}$ can be used to infer the probability that Tom likes the book given that Anna likes it and they have the same age, the probability that Anna likes the book given that she shares the same age with Tom and he likes the book, or the probability that Tom and Anna share the same age given that they both like the same book, etc.

The probability distribution over the possible worlds is:

$$\begin{aligned} P(X = x) &= \frac{1}{Z}\text{exp}(\sum_{i=1}^{F} w_i n_i(x)) \\ &= \frac{1}{Z}\prod_{i=1}^{F} \phi_i(x_{\{i\}})^{n_i(x)} \end{aligned} \tag{2.3}$$

where F is the number of formulas in the MLN, $n_i(x)$ is the number of true groundings of F_i in x, and $\phi_i(x_{\{i\}}) = e^{w_i}$. With increasing formula weights, an MLN increasingly resembles a purely logical KB.

Conceptually, inference in MLNs has two phases: the grounding phase, which constructs a large, weighted SAT formula, and the search phase, which searches for a low cost assignment (solution) to the SAT formula from grounding (using a satisfiability solver such as WalkSAT [KAUTZ et al. 1996]). We will base our approach on a state-of-the-art implementation, which offers scalable and efficient solutions for *maximum a posteriori* (MAP) inference and marginal inference.

Chapter 3

State of the Art

Recommender systems (RSs) are a subclass of information filtering systems, which seek to predict the preference that a user would give to an item and, accordingly, generate recommendations of items for the user. They present an alternative approach for information retrieval when compared to modern search engines, which have an explicit demand for the user to initiate search by formulating a specific search query [Baeza-Yates and Ribeiro-Neto 2011].

Recommender systems were initially defined by Resnick *et al.* [Resnick and Varian 1997] as systems where "people provide recommendations as inputs, which the system then aggregates and directs to appropriate recipients". In the present day, the definition of recommender systems has been extended to describe any system that generates personalized recommendations as output and/or guides the user in an individualized manner to interesting objects[1]. Recommender systems have been in increasing demand and an integral part of environments such as the Web, where there is a very large amount of information to be surveyed by the individual.

Recommender systems are based on information collected from the preferences of users to objects, which can be acquired implicitly or explicitly. Implicit preferences

[1] Adhering to RS literature, we use the term *object* and *item* interchangeably

are typically acquired by monitoring user's behavior, for example pages visited, songs heard, etc. Explicit preferences are typically collected in the form of users' ratings. Other information that can be used in RSs consists of user demographic data, such as nationality, gender, age, or data related to the attributes of the objects, for example type of object (book, song, clothes), author/provider, price, etc.

Various approaches have been proposed to generate recommendations. Among them, collaborative filtering (CF) techniques have played an important role, although they are very often applied in combination with other filtering techniques like content-based or social-based methods. Their combination leads to systems that are referred to as *hybrid recommenders*. CF methods are classified as memory-based and model-based. Memory-based approaches initially compute similarities of users or items by analyzing the user-item rating matrix. They then exploit these similarities to recommend items, considering preference similarity of like-minded users or similarity of rated items. Model-based approaches initially learn a prediction model using a training set that could be derived from the user-item rating matrix, and then apply the model to predict unknown preferences of users on items. A recent survey of RSs is presented in [BOBADILLA et al. 2013].

Besides combining content-based and CF-based methods to improve the performance of recommender systems by taking advantages of both perspectives, another trend has been the adoption of semantic technologies. It has been shown that the accuracy of recommendations can be improved by integrating Web site content and site structure, particularly by extending them with rich semantic data that characterize such content. Thereby, various approaches have been proposed to link semantic technologies and recommendations, leading to *semantic recommender systems* [PEIS et al. 2008].

The aforementioned systems focus on generating recommendations in a single domain, i.e. a system with distinct set of users and objects. Meanwhile, the Web as a complex environment with heterogeneous information has called for novel systems that encompass recommendations across domains. These are referred to as *cross-domain recommender systems* [WINOTO and TANG 2008]. Based on the way that they exploit the information across domains, they are further categorized as *collective* or *adaptive* systems.

	Semantic	Hybrid	Cross-domain	
[Zhou et al. 2004]	✓			
[Mobasher et al. 2003]				
[Shoval et al. 2008]				
[Maidel et al. 2008]				
[Adda et al. 2010]				
[Mabroukeh and Ezeife 2011]				
[Ruiz-Montiel and Aldana-Montes 2009]				
[Middleton et al. 2009]				
[Wang et al. 2009]				
[Wang 2011]				
[Codina and Ceccaroni 2010]				
[Codina and Ceccaroni 2012]				
[Eirinaki et al. 2006]	✓	✓		
[Jin et al. 2004]				
[Anand and Mobasher 2007]				
[Liu et al. 2007]				
[Wang and Kong 2007]				
[Ziegler et al. 2004]				
[Cantador et al. 2008a]				
[Cantador et al. 2008b]				
[Nguyen et al. 2010]				
[Senkul and Salin 2012]				
[Abel et al. 2013]			✓	Collective
[Szomszor et al. 2008]				
[Shi et al. 2011a]				
[Li et al. 2009c]				
[Pan et al. 2011]				
[Cremonesi et al. 2011]				
[Zhang et al. 2010]				
[Gao et al. 2013]				
[Nakatsuji et al. 2010]				
[Tang et al. 2011]				
[Kaminskas and Ricci 2011]			✓	Adaptive
[Berkovsky et al. 2008]				
[Winoto and Tang 2008]				
[Li et al. 2009a]				
[Pan et al. 2010]				
[Tang et al. 2012]				
[Wang et al. 2012]				
[Zhao et al. 2013]				
[Fernández-Tobías et al. 2011]	✓		✓	
[Loizou 2009]				

Table 3.1: Classification of recommender systems based on three aspects.

The work presented in this thesis lies at the intersections of *semantic* recommenders, *hybrid* recommenders, and *cross-domain* recommender systems. In this overview of the state of the art techniques, we precisely focus on systems that fall under these categories. In Table 3.1, we illustrate the classification of the approaches that are most relevant to our work. A few representatives from each category are described in the subsequent sections of the thesis.
As it can be observed in Table 3.1 and also argued below when describing these approaches, there is an obvious deficiency of hybrid approaches proposed in the scope of cross-domain recommender systems. At the same time, the adoption of semantic techniques has been barely investigated for cross-domain recommenders. The work in this thesis aims to fill particularly this gap by presenting cross-domain recommendation approaches that apply both hybrid methods and semantic technologies.

3.1 Semantic Recommender Systems

The systems generally classified as semantic (or semantically-enhanced) recommender systems adopt semantic technologies with respect to the representation of user and item information. They rely on a knowledge base usually defined as a concept diagram (like a taxonomy or thesaurus) or an ontology. These approaches exploit the semantic relations among attribute values of items in order to enhance traditional content-based RSs. Various works in semantic RSs have demonstrated that the exploitation of such semantic relations helps to improve performance of traditional models, especially in cold-start scenarios (i.e. settings with few available user preferences).
Earlier works [Zhou et al. 2004, Mobasher et al. 2003] present an adoption of semantic features to generate recommendations. The proposed approaches integrate user preferences implicit in their navigation data, i.e. a sequence of visited Web pages, with an ontological representation of the page content. It is demonstrated that an integrated approach yields significant improvements in terms of increasing the accuracy of recommendations. Regarding the semantic representation of the content, Zhou *et al.* [Zhou et al. 2004] restrict to a tree structure to store the access patterns, which is then used for matching and generating Web links for

recommendations. Whereas, Mobasher *et al.* [MOBASHER et al. 2003] use a flat (matrix-based) representation of the semantic data, without capturing structure in the semantic similarity measures.

Shoval *et al.* [SHOVAL et al. 2008] also propose the use of a common ontology to enable the description not only of items, but also users' profiles with concepts taken from the same vocabulary. Based on this representation approach and utilizing the ontology hierarchy, the authors introduce a content-based method to filter items for a given user. In a follow-up work [MAIDEL et al. 2008], the authors present an ontological content-based filtering method for ranking the relevancy of items in the domain of electronic newspapers.

A more recent line of works [RUIZ-MONTIEL and ALDANA-MONTES 2009, LOPS et al. 2009, WANG et al. 2009, WANG 2011, CODINA and CECCARONI 2010, CODINA and CECCARONI 2012] also incorporate semantics in content-based recommender systems. Codina *et al.* [CODINA and CECCARONI 2010] present an approach for recommendation in a movie domain, where item descriptions are based on semantic annotations referring to concrete ontology concepts. An extension of this work is presented in [CODINA and CECCARONI 2012] with new methods for measuring the semantic relatedness between attribute values of items based on their co-occurrence in similar contexts. In comparison to our work, these approaches focus only on generating recommendations a single domain.

3.2 Hybrid Recommender Systems

The evolution of recommender systems has witnessed the importance of combining different techniques, such as content-based and CF-based, in a hybrid approach to achieve peak performance. The proposed hybrid recommender systems may use various ways of merging the different techniques in order to gain advantage from them all with fewer drawbacks of any individual technique. Burke [BURKE 2002] presents a detailed survey focused on the hybrid systems.

There are many works that propose hybrid systems and this area in itself is broad. In order to position our work with respect to the most related works, we focus on those approaches that apply hybridization in combination with semantic technologies. Therefore, we restrict our scope to the hybrid semantic recommenders. We show in Table 3.1 the references to these works and describe below some of them proposed in different years.

In one of the earliest works, Eirinaki *et al.* [EIRINAKI et al. 2006] present a system for personalizing the recommendations to the users in a Web site by exploiting usage logs with semantics of the site content. This content is represented through an ontology, which is in fact only restricted to a taxonomical form (conceptual hierarchy). The work of Ziegler *et al.* [ZIEGLER et al. 2004] also addresses the use of semantic background knowledge about products and their taxonomy in order to improve the inference ability to establish user profiles. The approach particularly tackles the problem of preference feedback sparseness through a semantically-enhanced hybrid information filtering method, which allows to infer profile similarity between users even when they have not rated the same products.

Other approaches, like the one introduced by Cantador *et al.* [CANTADOR et al. 2008b], have been introduced to exploit the semantic-based knowledge representations for describing user profiles, in order to make enhanced and understandable recommendations. The approach assumes the availability of items that are semantically annotated by domain concepts (instances or classes) from an ontology-based knowledge base. The preferences of each user are then represented as a vector of weights, which measure the strength of the interest of user to the concepts in the ontology. The ontology-based user profiles are used to identify communities of interests through clustering of the profiles shared by users. Based on the clusters, interest networks can be built and used to generate recommendations. Hence, this strategy is applied in a content-based collaborative recommendation model, further presented in [CANTADOR et al. 2008a].

The semantically-enriched user profiles introduced in [ANAND and MOBASHER 2007] additionally incorporate in the recommendation process the user context, which is a combination of preference models from previous (long-term and short-term) user interactions.

Based on the assumption that a knowledge base of items already exists in the form of an ontology, the user preference model is represented as an ontological profile. Each item is related to a concept in the ontology, and a profile consists of a set of instances of the item ontology and a set of weights associated with the edges of the ontology. The weights denote the influence of the user ratings by the particular concepts. Another semantic-enhanced recommender system is introduced in Wang *et al.* [WANG and KONG 2007], extending the traditional clustering methods based on the user-item rating matrix with semantic information, in particular user demographical data and item category features, for computing user-pairs similarities.

More recently, Senkul *et al.* [SENKUL and SALIN 2012] propose a framework that integrates semantic information in the process of mining Web navigation patterns. The generated frequent navigation patterns are composed of ontology instances instead of Web page URLs. These navigation patterns are fed to a Web page recommendation mechanism. This mechanism not only serves to evaluate the pattern mining process, but most importantly it shows the increase in recommendation accuracy when navigation patters are leveraged with semantic information.

For the semantic enrichment of logs, most of the works assume the existence of already annotated Web pages. In [SENKUL and SALIN 2012], the mapping between ontology instances and the requested Web address in the Web server log is performed manually. An ontology is initially constructed based on the content and the database structure used in the Web site. Each page in the Web site is manually mapped to a concept in the ontology.

All these works propose solutions to making recommendations within a single distinct set of users and items, or to what we refer as a single domain. As such, the exploited ontologies are also extracted or engineered to model the content of a single site. Only recently, the community has shifted the interest on recommendation approaches across various domains. This is the focus of the next section.

3.3 Cross-domain Recommender Systems

Up until recently, the research community in recommender systems was particularly involved with techniques that focus in one particular domain of users and items. For example, the goal was to suggest relevant recommendations of books to users who have expressed preferences on books only. In terms of quality, their main objective is the improvement of accuracy of recommendations.

In the last years, we witness a growing interest to consider needs and preferences of users that span in different domains, and then generate recommendations of items that also belong to various domains. An example of a recommender system in this setting would be able to recommend books to a user who has only expressed music tastes. Such systems, referred to as cross-domain recommenders, are gaining increasing attention. This is of special interest in the heterogeneous open Web because of this desirable ability to suggest objects that do not necessarily belong to the domain where the user provided the ratings.

The survey of Fernandez [FERNÁNDEZ-TOBÍAS et al. 2012] presents an interesting classification of cross-domain recommendation techniques, extending previous classification schemes [BIN 2011, PAN et al. 2011]. Two main groups of cross-domain recommendation approaches are distinguished: *collective* and *adaptive*. Collective models exploit simultaneously the information from several domains and potentially generate joint recommendations for the domains. Adaptive approaches exploit in a directional manner the information from a source domain to a target domain, in order to improve then recommendations in the target domain.

One of the earliest studies of cross-domain recommendations is presented by Winoto *et al.* [WINOTO and TANG 2008], who ask the question *"If you like the Devil Wears Prada the book, will you also enjoy the Devil Wears Prada the movie?"* This work reports on a study conducted to discover association between user preferences on related object across different domains. The study aims to uncover two issues: whether there is cross-domain interest at a group level, and whether the preferences of a user at the individual level in one domain are useful for predicting preferences in other domains. A correlation analysis of the first issue finds out that in general there exists a trend of users' preferential behavior for items of different domains. They further indicate that cross-domain methods lead to higher diversity of items in the recommendations list, which does not compromise the performance of the systems. Additionally, it gives a significant benefit in increasing user satisfaction from this serendipity feature, which is much desirable for assuring higher utility of recommendations.

An alternative technique to integrate multi-domain user preferences consists of establishing relations between domain characteristics, and exploiting them for the cross-domain recommendation tasks. Approaches in this line [AZAK 2010, SHI et al. 2011b, CREMONESI et al. 2011, KAMINSKAS and RICCI 2011] integrate user preferences in multiple domains by establishing explicit relationships between domain characteristics. These relationships mainly exploit content-based features.

In [Cremonesi et al. 2011], the traditional relationships measuring the similarity in terms of user-based or item-based CF are modeled via a directed graph. Specifically, items of different domains are projected in one graph, then edges are created to connect the most similar items based on user ratings. These edges also act as bridges connecting the domains. This approach also requires several users to have rated items belonging to different domains. The graph-based structure is used to construct an extended adjacency matrix, which is used as conventionally to generate CF recommendations.

The idea of establishing bridges between the domains has attracted other researchers, who have proposed to use social tagging information for that purpose. Tags assigned by users to items act as an agreed terminology to describe them, even though they belong to different domains (e.g. tagging books and movies with similar emotional tags). Kaminskas *et al.* [Kaminskas and Ricci 2011] propose to exploit the tags to build item profiles, which are then used to compute their similarity. The most similar items tend to be recommended together. Shi *et al.* [Shi et al. 2011b] also exploit the tags to extend a matrix factorization approach to collaborative filtering by building user-user and item-item similarity matrices. In both these cases, it is assumed that there is information shared in both domains, such as in the form of tags.

Another different direction in cross-domain recommender systems has been the adaptive approach, where potentially useful knowledge is learned in one domain, it is transferred to a target domain where the preference feedback of users is sparse, and is exploited to enable better predictions there. We address these works in more details in Section 6.2.

As also highlighted in the survey [Fernández-Tobías et al. 2012], it can be well observed that hybrid cross-domain recommenders incorporating content with collaborative filtering are barely investigated. One reason could be the deficiency of datasets that make public both content-based and collaborative filtering information on several domains. Furthermore, there are only few techniques for cross-domain recommendations that address the use of semantic technologies [FERNÁNDEZ-TOBÍAS et al. 2011, LOIZOU 2009]. In Section 5.7, we give a detailed description of these approaches and compare them to our work.

The work in this thesis aims at delivering recommendation techniques that are 1) hybrid i.e. using both content-based and collaborative filtering features, 2) incorporate semantic representation of the content, and 3) generate recommendations not only in a single domain, but across different domains.

Thereby, this thesis particularly addresses the area that exists at the intersection of these three different groups of recommender systems. From the presented literature review and recent surveys, we have identified that there are no works standing at the intersection of the research areas. We are filling a gap of such systems that provide the three characteristics identified above.

Part II

Cross-domain Recommendations based on User Web Behavior

CHAPTER 4

Formalization of User Browsing Behavior

Records of Web usage behavior are produced daily in large amounts and the task of deriving actionable knowledge from these data is challenging. Investigations of user browsing behavior across different Web sites, not restricted to a single site, have shown to be beneficial, but they still need to tackle the problems of information heterogeneity and mapping usage logs to meaningful events from the application domain.

In this chapter, we focus on the problem of modeling user browsing behavior at multiple Web sites. We introduce a formal model to represent such usage behavior, the Web browsing Activity Model (WAM). We also present a formalization approach coupled with novel techniques for the semantic enrichment of usage logs with domain knowledge. These techniques are based on a combination of Semantic Web technologies and Machine Learning methods.

We demonstrate the feasibility and effectiveness of the implemented approaches through extensive experiments with real-world datasets of user browsing logs.

4.1 Introduction

Understanding the behavior of users in accessing Web resources is a powerful tool for Web sites providers to capture user navigation intentions, and accordingly improve their applications, build adaptive sites, or improve search. Furthermore, it helps to build recommendation systems that predict future user actions based on their past behavior.

There is an increasing body of literature on the investigation of clickstream data and navigation behavior modeling, with the majority focusing on data collected in a single site. The study from Park and Fader [PARK and FADER 2004], an initial inspiration for this work, shows the benefits of investigating user behavior at *multiple* Web sites[1]. This study and a few similar approaches argue on the significant added-value of investigating behavior at various sites in order to derive actionable behavioral knowledge and make better future forecasts. There has been a growing awareness that user interests and needs span across different application areas, leading to an emerging trend of of predictive solutions that offer cross-domain recommendations.

Despite the recent development and studies of user behavior in an open Web cross-domain setting, from a pragmatic standpoint the problem of the information heterogeneity encountered at different Web sites remains a challenge. We approach this problem at its root, addressing the task of formalizing and semantically-enriching user browsing behavior across different Web sites. Usage data, also referred to as usage logs, are syntactic representations of Uniform Resource Locator (URL) requests of Web resources accessed by the site visitors. Due to the primarily syntactical nature of such requests, comprehension of user browsing patterns is very limited. Hence, there is an urge for formalization approaches that leverage the semantics of the usage logs in accordance with the domain where they were tracked.

The goal of this work is to formalize user browsing behavior not only restricted at a single Web site, but rather at multiple sites. We investigate methods for mapping the records of usage logs issued by human visitors to meaningful events of the ap-

[1] The term Web *site* refers to the Pay-Level Domain (PLD). It allows us to identify a realm, where a single organization/user is likely to be in control. For instance, the PLD for *www.dbpedia.org* would be *dbpedia.org*.

plication domain where they were triggered. Hence, instead of a syntactic representation, we provide a semantic, formal description of usage logs by assigning them to concepts of a vocabulary from the respective domain knowledge, also referred to as contextualized domain knowledge. Most approaches use flat taxonomies to represent such vocabulary. In this work, we address the use of ontologies to structure domain concepts and relations, ensuring a richer semantic representation of a Web site content.

In Figure 4.1, we illustrate graphically how the work presented in this chapter is positioned with respect to the overall thesis framework. The proposed approach captures the process from the *acquisition* of user behavior in platforms such as Web servers or Web client programs (e.g. Toolbars), to the *formalization* of user browsing behavior models with description logic, and, furthermore, their *semantic enrichment* with contextualized domain knowledge.

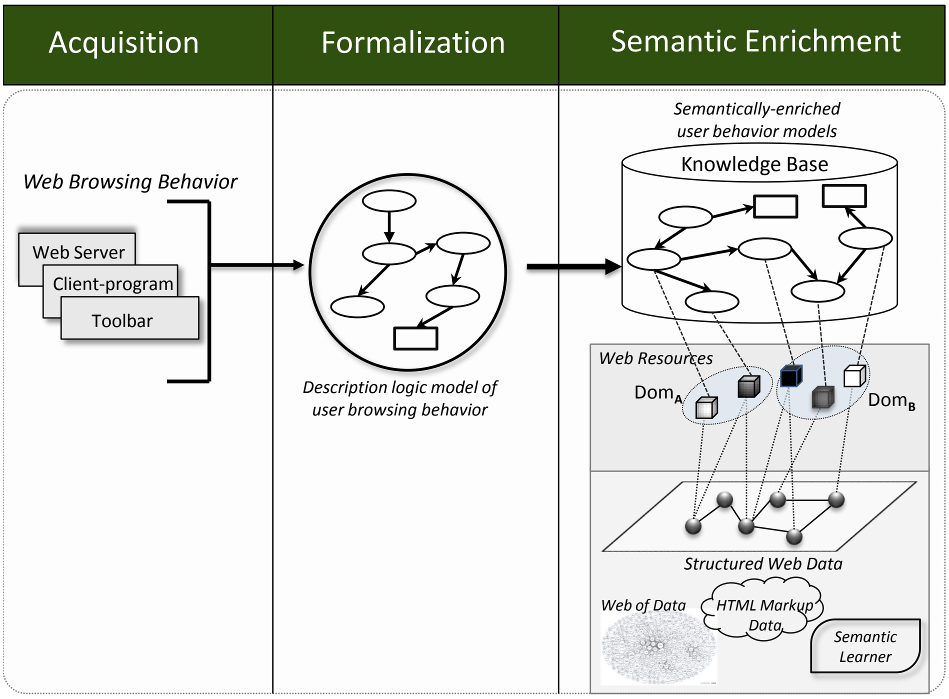


Figure 4.1: Framework overview: from acquisition to semantic enrichment of Web browsing behavior

There are several benefits from leveraging usage data with semantics through mappings to comprehensible events from the application domain. A key advantage is the capability to discover more insights about user behavior. Another benefit is the increased understandability of user behavior with respect to the application domain. It enables analysis at different levels of abstraction e.g. in a URL such as *<http://www.example.com/search?q=paris+louvre>* we could subsume parameters (*museum* instead of *Louvre*, or *city* instead of *Paris*). These subsumptions, helping to hide details of the accessed information, can be useful for privacy protection techniques.

Additionally, it allows formulation of more expressive queries for mining user behavioral patterns, as we show in an extension of our work to usage analysis in Appendix B.1.

Most importantly, enhancement of usage data with domain knowledge provides a rich context of user behavior that can be exploited for intelligent recommendation methods. In the following chapters of this thesis, we focus particularly on the task of designing recommendation approaches that use the semantically-rich user Web behavior model. Before investigating prediction and recommendation techniques, we initially commit to the modeling part, which is also the focus of this chapter.

4.1.1 Research Questions and Contributions

In this chapter, we address the following research question:

Research Question 1. *How can we model user Web browsing behavior with a formal representation, which semantically leverages the structured descriptions of resources in the Web?*

This research question comprises two main aspects. First, it addresses the problem of overcoming the shortcomings of syntactic representations of user browsing behavior data through a model that provides a formal and machine-readable representation. The second aspect consists in enriching this model with semantic descriptions, which capture contextualized knowledge of Web resources.

The addressed problem raises auxiliary questions of how to discover contextual knowledge from the application domain of the browsed Web resources, and how to achieve semantic enrichment in an open Web setting, without relying on a centralized knowledge base.

The investigation of these research questions led to the following main contribution:

Contribution I. *Formal model and semantic enrichment of user Web browsing behavior.*

This contribution is sustained through a theoretical model and artifacts, which we summarize below:

- We present a novel conceptual model for the formal representation of cross-site user browsing behavior. We introduce the Web browsing Activity Model (WAM), which enables a shared conceptualization of the knowledge from the various domains where the browsing logs are recorded (Section 4.2). This model is well-documented and published online.

- We propose a formalization and two-staged semantic enrichment approach comprising: (i) automatic technique to leverage the meaning of logs with domain knowledge available in the Web (Section 4.4), and (ii) supervised learning method to predict the semantic type of logs when it was not available in the earlier step, introducing a novel formulation of such problem as a multi-label classification task (Section 4.5).

- As an extension of our formalization approach, we present in Appendix B.1 a querying formalism and query answering method, which is used to find in the logs expressive patterns of usage behavior. This approach allows formulation of queries with temporal constraints, in addition to the semantic-based conditions.

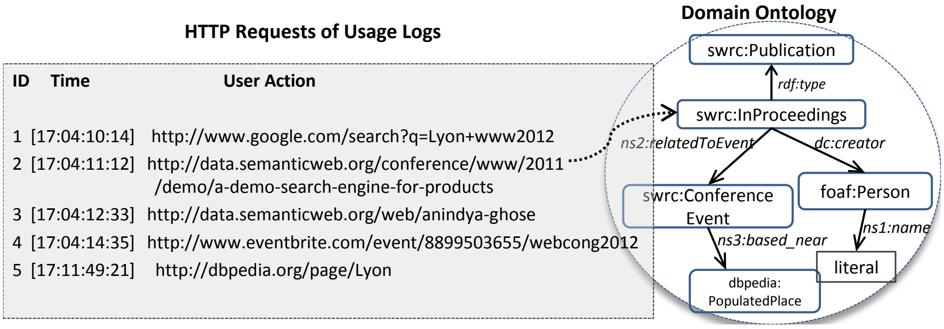
- Through evaluation experiments with real-world datasets from various Web sites, we demonstrate that the proposed formalization approach is feasible and the semantic enrichment techniques are effective in leveraging user browsing models with additional semantics from the extracted contextualized knowledge. The experimental setup and results of our evaluation are presented in Section 4.6.

4.2 WAM: Formal Model of Web Browsing Behavior

The benefits of usage behavior analysis have driven continuous research aimed at discovering navigation patterns in the logs issued by users while navigating the Web in the form of HTTP requests. Due to the primarily syntactical nature of these requests, user browsing logs lack a formal semantic representation. This makes the comprehension of the browsing patterns difficult.

A useful step towards a better interpretation and analysis of the usage patterns is to formalize the semantics of user browsing activity and the visited Web resources. We focus on this problem and present an approach for the semantic formalization of user Web browsing behavior. This model lays the foundation for effective techniques of querying expressive usage patterns, and for intelligent cross-domain recommendation methods.

When navigating the Web, users interact with Web resources via browser interfaces by clicking links, submitting HTML forms, etc. User behavior consists of a series of *browsing events* performed during these interactions. Traces of browsing behavior consists of sequences of Uniform Resource Locators (URLs) invoked at each interaction event. The information contained in the URL and the background knowledge in the underlying page can be used to characterize the different types of browsing events.


HTTP Requests of Usage Logs

```
ID   Time              User Action

1 [17:04:10:14]   http://www.google.com/search?q=Lyon+www2012
2 [17:04:11:12]  http://data.semanticweb.org/conference/www/2011
                   /demo/a-demo-search-engine-for-products
3 [17:04:12:33]  http://data.semanticweb.org/web/anindya-ghose
4 [17:04:14:35]  http://www.eventbrite.com/event/8899503655/webcong2012
5 [17:11:49:21]   http://dbpedia.org/page/Lyon
```

Figure 4.2: An example of user Web browsing logs and domain ontology

In the following, we present an example of an excerpt of logs issued by a particular user browsing the Web. We illustrate these logs in Figure 4.2, together with a part of the domain knowledge available in one of the visited sites in a structured representation. We refer to this structured representation of the domain knowledge as *domain ontology*.

Example 1. (Cross-site Browsing Logs and Domain Knowledge) *Suppose a user starts navigating the Web and the invoked URLs are recorded in a session of logs as illustrated in Figure 4.2. Initially, the user submits a search query at* google.com *engine with the keywords* Lyon *and* www2012. *The user continues by browsing a conference paper at the site* data.semanticweb.org *(SWDF)*[2]*, then visits the page of the paper's author. Afterwards, the user browses the page about* WebCongress2012 *at the site* eventbrite.com*, then visits the page about* Lyon *in the site* dbpedia.org.

In our approach, we map the entries of usage logs to concepts in the domain ontology. In this case, we illustrate a small part of the SWDF *domain ontology. The second log entry is linked to the resource of type* swrc:InProceedings [3]*, which is a subclass of the class* swrc:Publication. *We can extend the context for each log using the domain ontology, finding in this case additional knowledge (e.g. author, conference, location, etc.).*□

To provide a fine-grained description of Web browsing behavior, we use the term *event* to denote the basic component of user browsing behavior in interacting with the Web browser directly. In the following section, we introduce a set of definitions that capture different aspects of browsing behavior centered around the notion of *event*.

4.2.1 Definitions

Definition 6. (Event) *We define a browsing event as a tuple* $e_i = (l, \mathcal{T}, P, t)$*, where* l *is the full* URL *invoked,* $\mathcal{T}$ *is a set of* event types *for which the event qualifies,* P=$\{p_1, ..., p_m\}$ *is the set of* parameters *in the URL, and* t *is the* occurrence time. *For simplicity, we denote event time by* $e_i.t$ *and the set of event types by* $e_i.\mathcal{T}$*. The index* i *denotes the sequential* order *of events as they are issued in time.*

Each event that results from the interaction of a user with a specific Web page, serves a particular function (e.g. search within a portal, search in a search engine,

[2] Semantic Web Dog Food (SWDF)

[3] `http://ontoware.org/swrc/swrc_v0.3.owl#`

booking, login, etc.), and is related to a particular content (e.g. publication, city, organization, project, person, hotel, musician, etc.).

Definition 7. (Event Type) *An event e can be mapped to two types denoted by the set $\mathcal{T}$ = $\{\mathcal{T}_c, \mathcal{T}_f\}$, where $\mathcal{T}_c$ is the* type of content *to which this event relates and $\mathcal{T}_f$ is the* type of function *the event e serves.*

Each log entry is regarded as a browsing event. For example, referring to Figure 4.2, the last log entry is represented as e_5 = *(http://dbpedia.org/page/Lyon*, $\mathcal{T}$, P, *17:11:49:21)*, where content type $\mathcal{T}_c$=$\{$dbpedia:PopulatedPlace$\}$ and event function type $\mathcal{T}_f$ = $\{$wam:Informative $\}$. Other examples of content type for a browsing event include person, organization, real estate, education, stock exchange, etc. Examples of function type are login, search, checkout, reserve, etc.

We have extended the definition of a browsing event with the notion of parameter, which can be extracted from the URL l. Based on the typical convention of URL formation, we syntactically split the link into two basic parts: URL base, which refers to the pay level domain, and the rest of the URL is used to extract event parameters as tuples of variable name and value.

Definition 8. (Parameter) *An event parameter p is a pair p=(v_{name}, v_{value}) consisting of variable name and value. A parameter can be classified as an input or output parameter* [4].

In Example 1, the event e_1 has an input parameter of the form p*=(q, "Lyon+www2012")*. The events are further grouped into sessions, which represent a period of sustained Web usage. The boundaries of a session are normally determined by temporal and behavioral factors (e.g. browsing intention).

Definition 9. (Session) *A user session is a tuple $S = \{s, T_s, T_e, U\}$, where $s = \langle e_1, e_2, ..., e_n\rangle$ is an ordered sequence of browsing events performed by the user with identifier U, such that $e_i.t \leq e_{i+1}.t$ for all i, where i denotes the event order in the sequence. Furthermore, T_s is the starting time and T_e the ending time of the session, such that $T_s \leq e_i.t \leq T_e$.*

[4] In this work, we focus on the input parameters only.

The browsing events in Example 1 are grouped in one session, $S_1 = \{\langle e_1, e_2, e_3, e_4, e_5\rangle, e_1.t, e_5.t, 1\}$, where $e_1.t$=*"17:04:10:14"*, and $e_5.t$ =*"17:11:49:21"*.

Our goal is to map a browsing event to an information resource in the Web, which is identified in the content of the page underlying the event URL. As such, we give the definition of a Web resource.

Definition 10. (Semantic Web Resource) *We define as a Web Resource an information resource from the document Web that is identified by a dereferencable HTTP URI (Uniform Resource Identifier)*[5]*, and has an RDF/XML representation, which contains associated description of its properties and relations to other Web resources.*

Mapping a browsing event to a Web resource helps us to leverage the event with additional semantics, such as the type and semantic properties of the resource available in its RDF representation.

4.2.2 Model Requirements

In order to provide a formal model of user Web browsing behavior based on the given definitions, we take into consideration the following model requirements:

Requirement 1. Formal Semantics

The model serves as a common vocabulary to share information related to user browsing behavior in the Web. It expresses specific meaning of the modeled concepts and their relationships, providing an unambiguous interpretation. The model is not restricted to a taxonomy or enumeration of concepts, rather it offers higher expressiveness of concepts with different types of relations, properties and restrictions.

Requirement 2. Readibility

The model is offered in a structured representation and is machine-readible, allowing easier transfer and interoperability of data. The model is furthemore well-documented, and accessible to humans and machines.

[5] `http://www.w3.org/Addressing/URL/URI_Overview.html`

Requirement 3. Reasoning
The model is expressed using a formalism that enables automatic inferencing. It enables deriving logical consequences of the represented knowledge, i.e. new knowledge from the implicit knowledge expressed in the model.

Requirement 4. Reusability
The model facilitates the reuse of other structures of organized knowlege, in order to interoperate with other ontologies/controlled vocabularies. This permits us to leverage other people's ontology building work, using previously validated and authoritative source ontologies.

Requirement 5. Extensibility
The model can be easily extended with concepts not yet captured, e.g. information related to user profiling or demographic properties, application-oriented data. etc. It should allow extensibility with knowledge that would be necessary for other applications, besides querying and prediction.

These requirements guide our engineering choices for the model introduced in the following section.

4.2.3 Web Browsing Activity Model

We present a formal model to capture browsing behavior of users in the Web, referred to as the **W**eb browsing **A**ctivity **M**odel (WAM). It is provided in an ontological representation, well-documented[6] and published online[7] in a machine-processable format.
WAM ontology is expressed using OWL-DL formalism with underlying $\mathcal{SHOIN}(\mathbf{D})$ description logic. In the following, we describe in details the components of the ontology using $\mathcal{SHOIN}(\mathbf{D})$ axioms.

Namespace. A standard initial component of an ontology includes the namespace declaration, as a means to interpret identifiers unambiguously and increase readi-

[6] `http://ais.al/ns/wam`

[7] `http://greenlinkeddata.org/wam.owl`

bility of the ontology presentation. The namespace of the WAM ontology we propose is declared with the prefix wam and URI *http://greenlinkeddata.org/wam.owl*.

Concepts in TBox. The first step in developing the basic structure of the ontology is the design of the TBox. The goal here is to establish a good foundation, which can be later extended without leading to inconsistencies.

In WAM, we distinguish three groups of concepts[8]: Core, External, and Type concepts. Core concepts are atomic concepts, which are defined in TBox using unique names and assumed to exist in the absence of any definitional axioms. These concepts in WAM are wam:Event, wam:Session, wam:User, wam:Parameter.

External concepts are classes imported from well-established ontologies using the subsumption relation. We reuse the *Event Ontology* [9] such that our central concept wam:Event is a subclass of event:Event. This is expressed in axiom 4.1.
The concept wam:Event has two subclasses wam:StartEvent and wam:EndEvent (axioms 4.2 and 4.3), which mark the start and end of a session, respectively.

$$\mathsf{wam{:}Event} \sqsubseteq \mathsf{event{:}Event} \tag{4.1}$$

$$\mathsf{wam{:}StartEvent} \sqsubseteq \mathsf{wam{:}Event} \tag{4.2}$$

$$\mathsf{wam{:}EndEvent} \sqsubseteq \mathsf{wam{:}Event} \tag{4.3}$$

We have designed a set of classes, which represent types of individuals. The most general of these concepts is wam:Type, which then subsumes wam:EventType and wam:ParameterType (axioms 4.4-4.5). Based on the definition of event types, a browsing event may have a function type and a content type. The classes that represent these types in WAM are wam:ContentType and wam:FunctionType (axioms 4.6-4.7). We model them as disjoint concepts (axiom 4.8). Partitioning the domain using disjointness allows reasoning tasks to be performed faster, since in

[8] The terms *concept* and *class* are used interchangeably

[9] `http://purl.org/NET/c4dm/event.owl`

DL concepts are assumed to overlap unless it is stated otherwise.

$$\mathsf{wam{:}ParameterType} \sqsubseteq \mathsf{wam{:}Type} \tag{4.4}$$

$$\mathsf{wam{:}EventType} \sqsubseteq \mathsf{wam{:}Type} \tag{4.5}$$

$$\mathsf{wam{:}ContentType} \sqsubseteq \mathsf{wam{:}EventType} \tag{4.6}$$

$$\mathsf{wam{:}FunctionType} \sqsubseteq \mathsf{wam{:}EventType} \tag{4.7}$$

$$\mathsf{wam{:}FunctionType} \sqcap \mathsf{wam{:}ContentType} \sqsubseteq \bot \tag{4.8}$$

The ontology also contains the concept wam:Parameter, which models the class of parameters contained in the URL of the log. This concept subsumes two other classes wam:InputVariable and wam:OutputVariable, which are disjoint. These concepts are summed up in axioms 4.9 and 4.11:

$$\mathsf{wam{:}InputVariable} \sqsubseteq \mathsf{wam{:}Parameter} \tag{4.9}$$

$$\mathsf{wam{:}OutputVariable} \sqsubseteq \mathsf{wam{:}Parameter} \tag{4.10}$$

$$\mathsf{wam{:}InputVariable} \sqcap \mathsf{wam{:}OutputVariable} \sqsubseteq \bot \tag{4.11}$$

Roles and Restrictions in TBox. Relations that hold among various individuals are defined with roles. Furthermore, role restrictions are used to assert how individuals of particular concepts are related via roles. Relations are asserted using existential and the universal role restrictions. WAM contains a set of roles for which we additionally define domain and range restrictions. They, thereby, restrict a given role in taking particular concepts as its domain and range, respectively.

Each event belongs to one session only, it has one timestamp, one order. It can have more than one wam:EventType, which is expressed in axiom 4.12 using the existential restriction. The relation wam:type can be either wam:contentType or wam:functionType (axiom 4.16).

An event has strictly one full URL and one base URL. Axiom (4.13), for instance, limits the range of wam:hasURL to only individuals of type wam:EventURL. When available, it can be related to the URI of Web resources.

Additionally, an event can have several (or none) parameters, which are restricted to individuals of class wam:Parameter. Each parameter has one value and one name. These roles and restrictions are summed up in the following axioms:

$$\begin{aligned}
\mathsf{wam{:}Event} \sqsubseteq\ & =_1 \mathsf{wam{:}belongsTo.wam{:}Session} && (4.12)\\
& \sqcap\ \exists\ \mathsf{wam{:}type.wam{:}EventType} &&\\
& \sqcap\ =_1 \mathsf{wam{:}order.order} &&\\
& \sqcap\ =_1 \mathsf{wam{:}hasURL.wam{:}EventURL} &&\\
& \sqcap\ \forall \mathsf{wam{:}hasURI.literal} &&\\
& \sqcap\ =_1 \mathsf{wam{:}hasTime.time{:}DateTimeDescription} &&\\
& \sqcap\ \forall \mathsf{wam{:}hasParameter.wam{:}Parameter} &&\\
\mathsf{wam{:}EventURL} \sqsubseteq\ & =_1 \mathsf{wam{:}fullUrl.literal}\ \sqcap\ =_1 \mathsf{wam{:}baseUrl.literal} && (4.13)\\
\mathsf{wam{:}Resource} \sqsubseteq\ & \forall \mathsf{wam{:}URI.literal} && (4.14)\\
\mathsf{wam{:}Parameter} \sqsubseteq\ & =_1 \mathsf{wam{:}hasName.wam{:}ParameterName} && (4.15)\\
& \sqcap\ =_1 \mathsf{wam{:}hasValue.literal} &&\\
\mathsf{wam{:}type} \sqsubseteq\ & \mathsf{wam{:}contentType}\ \sqcup\ \mathsf{wam{:}functionType} && (4.16)
\end{aligned}$$

Each session has at least one event, strictly one start event and one end event, one interval, and one user (axiom 4.17). For event timestamps and session intervals, we reuse basic concepts from *OWL Time* ontology [10] that models knowledge about time, such as temporal units, instants, etc. We reuse the classes time:DateTimeDescription, time:Instant, time:Interval, and time:TemporalEntity from this ontology.

We declare wam:hasEvent as an inverse role to wam:belongsTo (axiom 4.18). Inverse roles serve to express the opposite relation between two individuals. As such, wam:hasEvent could be used to relate a session to an event, while wam:belongsTo could be used for the inverse.

[10] http://www.w3.org/2006/time#

$$\begin{aligned}\mathsf{wam{:}Session} \sqsubseteq\ & \geq_1 \mathsf{wam{:}hasEvent.wam{:}Event} \\ & \sqcap =_1 \mathsf{wam{:}hasUser.wam{:}User} \\ & \sqcap =_1 \mathsf{time{:}interval.time{:}Interval} \\ & \sqcap =_1 \mathsf{wam{:}hasStartEvent.wam{:}StartEvent} \\ & \sqcap =_1 \mathsf{wam{:}hasEndEvent.wam{:}EndEvent}\end{aligned} \tag{4.17}$$

$$\mathsf{wam{:}hasEvent} \equiv \mathsf{wam{:}belongsTo}^{-1} \tag{4.18}$$

Another concept of our ontology is wam:User, which is simply characterized in WAM by the *Internet Protocol (IP) address* and strictly one identifier *userID* (axiom 4.19). The ontology allows flexible future extendability with user profiles or other attributes (e.g geographical information based on the IP, etc.).

$$\begin{aligned}\mathsf{wam{:}User} \sqsubseteq & =_1 \mathsf{wam{:}userID.literal} \\ & \sqcap \exists \mathsf{wam{:}userIP.literal}\end{aligned} \tag{4.19}$$

Individuals in ABox. The next step is to instantiate the various concepts and, thereby, populate the ontology by axioms about individuals. The set of such axioms is also referred to as the ABox component of the knowledge base. This step includes enumerating the individuals, sorting them, and finally asserting knowledge about each individual through assertion axioms.

We illustrate below a set of axioms asserting knowledge about the user logs shown in Example 1. An obvious first assertion is to instantiate the knowledge about user to which the logs belong. Thereby, we create an individual of type wam:User using axiom 4.20 and instantiate its userID in axiom 4.21. In a similar way, the ABox is further populated with other individuals related to session, events, and other time-related information as illustrated in the following axioms 4.20 - 4.43:

$$\mathsf{wam{:}User}(\mathsf{user1}) \tag{4.20}$$

$$\mathsf{wam{:}userID}(\mathsf{user1}, 1) \tag{4.21}$$

$$\mathsf{wam{:}Session}(\mathsf{sid1}) \tag{4.22}$$

$$\mathsf{wam{:}StartEvent}(\mathsf{e1}) \tag{4.23}$$

$$\mathsf{wam{:}order}(\mathsf{e1}, 1) \tag{4.24}$$

$$\mathsf{wam{:}hasStartEvent}(\mathsf{sid1}, \mathsf{e1}) \tag{4.25}$$

$$\mathsf{wam{:}EventURL}(\mathsf{e1url}) \tag{4.26}$$

$$\mathsf{wam{:}baseURL}(\mathsf{e1url}, \textit{google.com}) \tag{4.27}$$

$$\mathsf{wam{:}Event}(\mathsf{e5}) \tag{4.28}$$

$$\mathsf{wam{:}order}(\mathsf{e5}, 5) \tag{4.29}$$

$$\mathsf{wam{:}hasEndEvent}(\mathsf{sid1}, \mathsf{e5}) \tag{4.30}$$

$$\mathsf{wam{:}EventURL}(\mathsf{e5url}) \tag{4.31}$$

$$\mathsf{wam{:}fullURL}(\mathsf{e5url}, \textit{http://dbpedia.org/page/Lyon}) \tag{4.32}$$

$$\mathsf{wam{:}hasContentType}(\mathsf{e5}, \mathsf{dbpedia{:}PopulatedPlace}) \tag{4.33}$$

$$\mathsf{time{:}DateTimeDescription}(\mathsf{e5timedesc}) \tag{4.34}$$

$$\mathsf{wam{:}hasTime}(\mathsf{e5}, \mathsf{e5timedesc}) \tag{4.35}$$

$$\mathsf{time{:}unitType}(\mathsf{e5timedesc}, \mathsf{time{:}unitSecond}) \tag{4.36}$$

$$\mathsf{time{:}day}(\mathsf{e5timedesc}, \mathsf{20}) \tag{4.37}$$

$$\mathsf{time{:}month}(\mathsf{e5timedesc}, \mathsf{11}) \tag{4.38}$$

$$\mathsf{time{:}year}(\mathsf{e5timedesc}, \mathsf{2012}) \tag{4.39}$$

$$\mathsf{time{:}hour}(\mathsf{e5timedesc}, \mathsf{17}) \tag{4.40}$$

$$\mathsf{time{:}minute}(\mathsf{e5timedesc}, \mathsf{11}) \tag{4.41}$$

$$\mathsf{time{:}second}(\mathsf{e5timedesc}, \mathsf{49}) \tag{4.42}$$

$$\mathsf{time{:}timeZone}(\mathsf{e5timedesc}, \mathsf{tz} - \mathsf{us{:}EST}) \tag{4.43}$$

Figure 4.3: A set of assertion axioms in WAM ontology

Serialization. WAM ontology is also provided in the OWL RDF/XML syntax. The complete representation of the model in this format is available online[11].

Graph representation. In Figure 4.4, we illustrate a graph representation of our model. For better readibility, we have represented the classes of the ontology as nodes with different colors, distinguishing in this way the core classes, external classes, and type classes.

As mentioned above, the central concept is wam:Event, which represents the basic component of user Web browsing beviour. Roles are illustrated through the directed edges connecting the concepts. The direction of each edge specifies accordingly the domain and range of the property. For example, the class wam:Event is connected to wam:Session with the property wam:belongsTo. Inheritence is denoted through a pointed triangle edge, showing for example that wam:Event is a subclass of the imported class event:Event.

This ontology represents a core model for expressing the information contained in the Web browsing logs in a formal and semantic way. It can be extended with further concepts and properties depending on the particular domain and application under consideration. As such, if for a specific application we need to cover more information on user profiling, then we can extend WAM with other concepts and constraints related for example to user demographic or geographic data. This model serves a medium for shared understanding and representation of knowledge on user Web browsing activity.

[11] `http://greenlinkeddata.org/wam.owl`

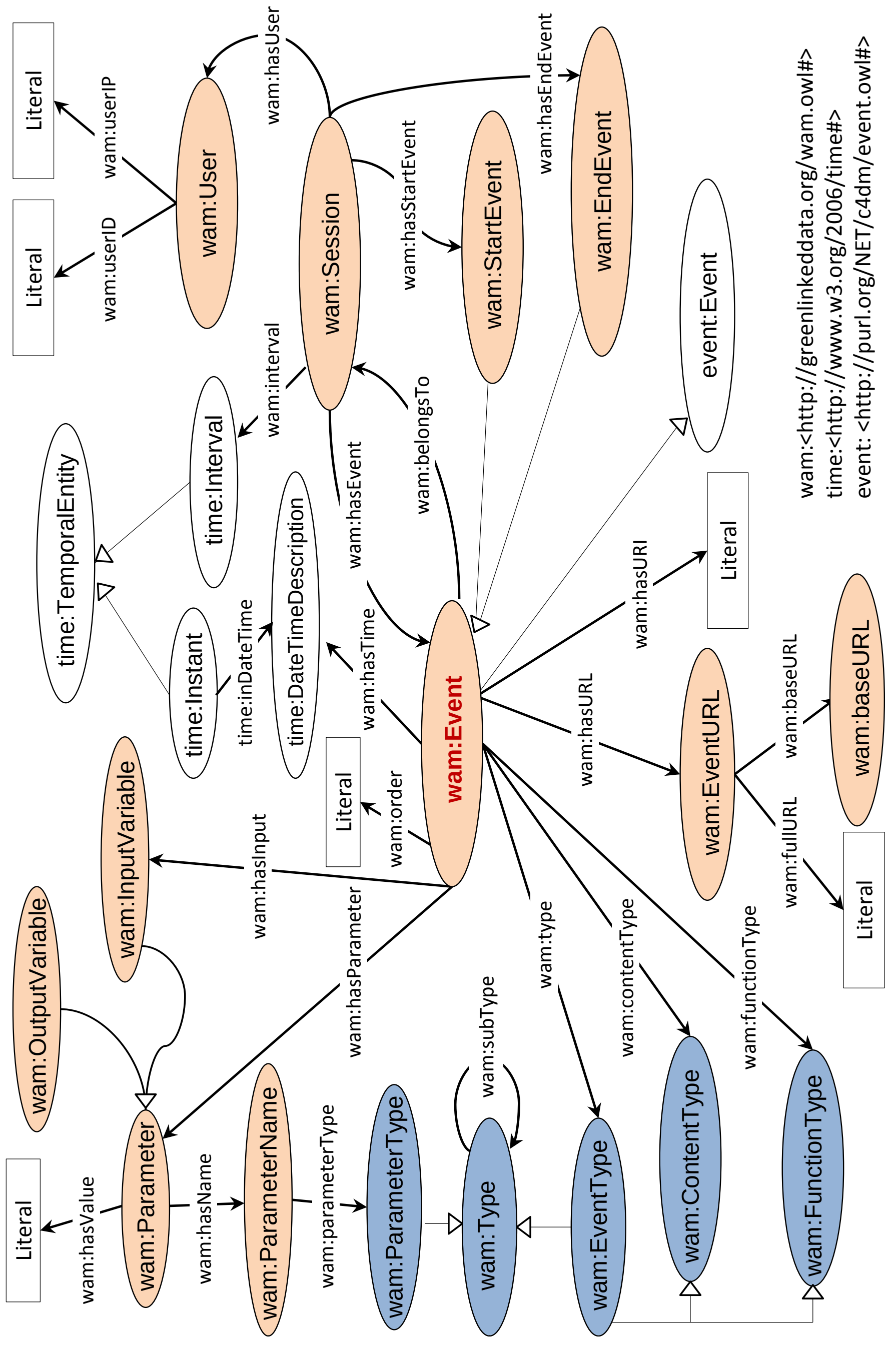


Figure 4.4: Web browsing Activity Model (WAM) - Graphical Representation

4.3 Formalization and Enrichment Process

User browsing behavior is usually captured by Web logs that are tracked in the respective servers of Web content providers or via client-side toolbars that users have installed and agreed to collect their data. The original browsing data logs are initially stored in raw form, as produced upon each user interaction with the Web browsers. Our formalization approach comprises a pipeline of techniques, starting with the preprocessing of logs and leading to the formal description of user browsing models based on the proposed ontology WAM. Furthermore, the formalized models are semantically enriched with additional background knowledge. We propose a set of techniques for such semantic enrichment. Figure 4.5 graphically illustrates the pipeline of the steps performed during the formalization and enrichment process.

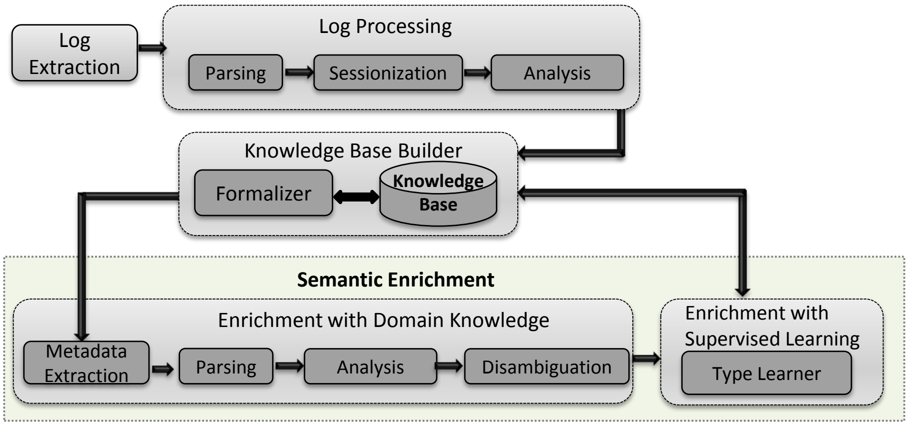


Figure 4.5: Pipeline of formalization and enrichment process

As shown in Figure 4.5, we start the formalization approach by extracting human-generated logs in a large corpus of available Web browsing logs. We further proceed with the parsing techniques.

Parsing. Different logging platforms (e.g. toolbars, Web servers) may deploy different formats for representing user browsing logs. As such, we deploy a generic parser based on rules, which can be configured for the format of the provided logs. Most logs have a particular structure, for example each line is composed of fields such as URL, HTTP method, client IP address, User Agent, User ID, etc. The User Agent identifies the HTTP client, containing information on browser and operating system. The User ID are not always available in logs. They are assigned independently for each visitor by the Web site implementation. In order for each of these fields to be extracted, we provide a set of parsing rules consisting of a collection of regular expressions. The parsing rules identify specific types of extracted fields and corresponding delimiters.

Sessionization. This step consists in grouping log entries into distinct viewing sessions based on characteristics that identify a particular user and the time of the page view. The rationale is that those page visits performed closer in time are considered to be a distinct session. Our sessionization algorithm interprets a unique combination of session timeout, User Agent, IP address and, when available, User ID as the start of a new session. A session timeout is the time of inactivity before a session is considered complete. Hence, we create a grouping for the logs of an identified user upon detecting a sizeable pause in time, for example a default heuristic is a *30 minutes* timeout between the events [BUCKLIN and SISMEIRO 2003, BERENDT et al. 2003].

Analysis. In order to gain insight about the logs, we have devised a set of methods that analyze the data and yield various statistics such as average length of browsing sessions, number of sessions, mode of session lengths, etc.

Formalization. After having extracted the necessary fields from each log entry and segmented the logs into sessions, the next step consists in constructing a formal and machine-processable description of these data. We make use of the ontological Web Browsing Activity Model (WAM). At this stage, with the axiomatization of the TBox in place, we need to instantiate the various concepts developed in the ontology and, accordingly, assert knowledge about individuals by creating ABox assertions.

This procedure is presented in Algorithm 1. It takes as input a string-based representation of the ordered browsing events in a session, and produces a set of ABox assertions related to each event based on the TBox concepts specified in the WAM. For simplicity of reading, we remove the WAM namespace in the algorithm, i.e. use term instead of wam:term.

```
Algorithm 1: Formalization of Web Browsing Events
  Input: Web browsing session $S = \{s, T_s, T_e, U\}$, s.t. $s = \langle e_1, e_2, ..., e_n \rangle$ is the
      ordered sequence of events, $U$ user identification, $T_s$ is the starting time and
      $T_e$ ending time of session $S$.
  Output: ABox assertions $\mathcal{A}$
   1: $\mathcal{A} = \emptyset$
   2: $j = 1$
   3: for all $e_i \in s$, do
   4:     $e_i$=($l$, $\mathcal{T}$, $\mathcal{P}$, $t$) with URL $l$, initially empty set of class types $\mathcal{T}$ and
          parameters set $\mathcal{P}$, and event occurrence time $t$
   5:     $b$ = extractBaseUrl($l$)
   6:     Assertions $\alpha_e$ = { Event($e_i$), EventURL($e_i\_url$), hasURL($e_i$, $e_i\_url$),
          fullURL($e_i\_url$, $e_i.l$), baseURL($e_i\_url$, $b$), hasTime($e_i$, convertTime($e_i.t$)),
          order($e_i$, $j$++)}
   7:     //Parameter-related assertions
   8:     $\mathcal{P}$ = extractParameters($l$);
   9:     for all $p_i \in \mathcal{P}$ s.t. $p_i = (v_{name}, v_{value})$ do
  10:         $\alpha_p$ = {InputVariable($p_i$))}
  11:         $\alpha_p$ = $\alpha_p \cup$ { ParameterName($p_i.v_{name}$),
              hasName($p_i$, $p_i.v_{name}$), hasValue($p_i$, $p_i.v_{value}$)}
  12:     end for
  13:     //Resource assertions
  14:     Resource assertions with Web of Data $\alpha_r$=assertSemResWB($e_i$, $b$)
  15:     if ($\alpha_r$ == $\emptyset$ ) then
  16:         Assertions with HTML Markup Data $\alpha_r$=assertSemResMD($e_i$, $b$)
  17:     end if
  18:     $\mathcal{A} = \mathcal{A} \cup \alpha_e \cup \alpha_r \cup \alpha_p$
  19: end for
  20: //Session-related assertions
  21: $S_{id}$ = sid($T_s$, $U$)
  22: $\alpha_s$ = { Session($S_{id}$), User($user\_U$), userID($user\_U$, $U$),
      hasUser($id_S$, $user\_U$), hasStartEvent($S_{id}$, $e_1$), hasEndEvent($S_{id}$, $e_n$)}
  23: $\mathcal{A} = \mathcal{A} \cup \alpha_s$
```

For each event of the session, we initially extract the base URL that corresponds to the pay level domain (PLD) (Alg. 1, line 5). Afterwards, we create a set α_e of event-related ABox assertions (line 6) and a set α_p of parameter-related assertions. From the URL of the event, we extract the parameters (line 8), each of them being a pair of variable name and variable value. The parameter is modeled as an instance of the class wam:InputVariable. Additional assertions on wam:ParameterName, wam:hasName, and wam:hasValue are created (line 9-12).

A crucial part of our approach is the step of mapping a browsing event to a representative Web resource in the application domain, i.e. its respective Web domain b, and accordingly assert additional domain knowledge in the set of assertions α_r (line 14-17). This is performed in the semantic enrichment procedures $assertSemResWB$ and $assertSemResMd$, which we describe in Section 4.4.

In the next step, we instantiate knowledge related to the session of the events. For each session, we create a unique ID with the procedure $sid(S)$ using a combination of session start time T_s and user identifier U (line 21). The ABox assertions created about the session (line 22) are additionally added to the overall set of assertions $\mathcal{A}$ of our knowledge base.

Knowlegde Base. The product of the aforementioned knowledge engineering steps is a Knowledge Base, which consists of the TBox axioms of the WAM ontology and the constructed ABox assertions about the browsing events and sessions. The knowledge is serialized in RDF format and stored in a repository. The use of an RDF repository enables manipulation, querying, and reasoning about the data with standard tools, such as OWL reasoners and graph-based languages, such as SPARQL query language[12] for RDF triplestores.

The formalized knowledge in the repository constitutes semantic-rich, machine-processable models of user Web browsing behavior. It lays the foundation for intelligent techniques that exploit this knowledge in diverse ways, such as for Web behavior analysis via expressive querying and pattern mining, or semantic-based behavior prediction techniques.

Semantic Enrichment. Each Web browsing event is intentionally performed by the user to serve a particular function and it relates to a specific content. Our goal is to discover such content that potentially comprises contextualized knowledge in the respective domain where the event occurred. Semantic enrichment of browsing behavior models is a specific meta-data generation process, which aims to couple

[12] `http://www.w3.org/TR/rdf-sparql-query/`

each of the formalized browsing events with additional contextualized knowledge in structured form.

The semantic enrichment approach that we propose consists of two stages. The first stage comprises a set of techniques that harvest the domain knowledge available in a structured format in the Web, and use it to semantically-leverage the formalized browsing behavior models (Section 4.4).

The second stage tackles the case when such structured knowledge is not available for the domains of interest where the browsing behavior was tracked (Section 4.4.2). We provide a learning technique that is able to learn in a supervised way the missing semantic content type of the browsing event, which refers to the wam:ContentType class in WAM.

4.4 Semantic Enrichment with Domain Knowledge

In the first stage of our approach, we enhance the formalized behavior model by harvesting the structured data available in the Web. More and more Web publishers are providing their data online coupled with meta-data to express the semantic meaning of the content. There are two main classes of meta-data pervasive today.

The first class is based on a set of best practices for publishing and connecting data known as Linked Data. They underpin the concept of Semantic Web, which consists in extending the Web from a collection of mainly human-readable HTML documents into a global space that connects data from diverse domains. This global space, coined the *Web of Data*, is explained as ”a web of things in the world, described by data on the Web” [Bizer et al. 2009].

The second class consists of semantic meta-data embedded within the HTML content of Web pages. The authors mark up human-readable information with machine-readable data that allows agents to interpret them. This class is referred to as *structured markup data* embedded in HTML.

We exploit both classes of semantic data, i.e. the Web of Data and structured markup data that are available in the Web. In the following sections, we describe our methodology in harvesting these data to semantically enrich user browsing behavior models. Our approach is open-domain in that it does not exploit domain-specific heuristics.

4.4.1 Semantic Enrichment using the Web of Data

Recent years have marked an increasing adoption of Linked Data principles that has led to a growing Web of Data. There are more and more semantically-enabled Web sites, which provide a semantic representation of their content and respective Web resources, together with links to data in other domains. This dataspace, also known as the Linked Data Cloud, offers an abundance of datasets with structured knowledge in RDF/XML representation.

For this dataspace to come to fruition, ontological support plays a seminal role in enabling information exchange across distributed datasets. Formal ontologies provide shared vocabularies that help to better specify and align the terminology used in different datasets.

In our work, we harvest the Web of Data in order to enrich with additional context the formalized user browsing behavior models. More precisely, our goal is to map each browsing event to semantic resources from the application domain. We further explore the structured content of this resource, exposed in the Web of data, in order to expand that browsing event with more contextualized knowledge from the respective domain. In Algorithm 2, we describe the procedure to establish this mapping and, accordingly, retrieve the structured representation of the referenced resource.

Algorithm 2: Resource assertions *assertSemResWB*(e_i,b) using Web of Data

```
Input: Event $e_i$ and its corresponding Web domain $b$
Output: Set of ABox assertions $\alpha_r$
//Web Resource Identification
RDF document $\mathcal{D}$=dereferenceURL($e_i.l$)
Web Resource URI $R_l$=identifyResourceURI($e_i.l$, $\mathcal{D}$)
Assertion $\alpha_r$ = { hasURI($e_i$, $R_l$) }
//Semantic Types
Domain ontology $\mathcal{O}^b$ = getDomainOntology(b)
Find classes $\mathcal{T}_r$=resourceClassification($R_l$, $\mathcal{O}^b$)
for all $T \in \mathcal{T}_r$ do
    $\alpha_r = \alpha_r \cup \{ T \sqsubseteq \mathsf{ContentType} \}$
    $\alpha_r = \alpha_r \cup \{ \mathsf{contentType}(e_i, T) \}$
end for
```

In order to get more information about the resource under the URL $e_i.l$, we apply *URI Dereferencing*. This is the process of looking up a URI on the Web in order to get information about the referenced resource. Our goal is to locate the structured content of this resource, namely its RDF representation. We use a HTTP mechanism called *content negotiation*[13] [14].

Using content negotiation, HTTP clients can specify in the header the kind of representation they require, for example in case the client prefers HTLM, then the server generates an HTML representation. We send a HTTP request on the given URL specifying in the header that we require a RDF representation. Technically, we perform an HTTP GET request and send an *Accept: application/rdf+xml* header along with the request. The server sends us back a RDF/XML document containing the description of the original resource.

In the case when we are not able to find a RDF document after dereferencing the URL, we deploy the technique presented next in Section 4.4.2, which extracts the structured content embedded in HTML pages.

When dereferencing is successful, we parse the RDF document and locate the resource that this document describes, identifying it with the respective URI (Alg. 2, line 2). This information is extracted from the RDF statement containing the rdf:about property. For example, the RDF document retrieved by dereferencing the URL <*http://dbpedia.org/page/Lyon*> describes the resource with URI <*http://dbpedia.org/resource/Lyon*>.

Resource Classification. In the next step, we find the semantic types of this resource expressed in RDF statements with the rdf:type property (lines 4-5). A resource type is denoted by the class to which it belongs in the domain ontology. This ontology O^b can be accessed at the URI identified by the namespace in the retrieved RDF document.

It may be the case that one resource is expressed as an instance of multiple classes in O^b. For example, the DBpedia resource of *Lyon* may be an instance of class dbpedia-owl:Settlement and class dbpedia-owl:Place, where dbpedia-owl is the

[13] `http://www.w3.org/Protocols/rfc2616/rfc2616-sec12.html`

[14] `http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/LinkedDataTutorial/`

namespace of the DBpedia ontology[15]. In our knowledge base, we model these classes as subclasses of wam:ContentType, and accordingly establish the relation wam:hasContentType between each of these classes and the browsing event (Alg. 2, line 6-9).

We illustrate in Figure 4.6 an example of semantically enriching an event with contextual knowledge from the Web of Data. This event corresponds to the second entry of logs shown in our running Example 1.

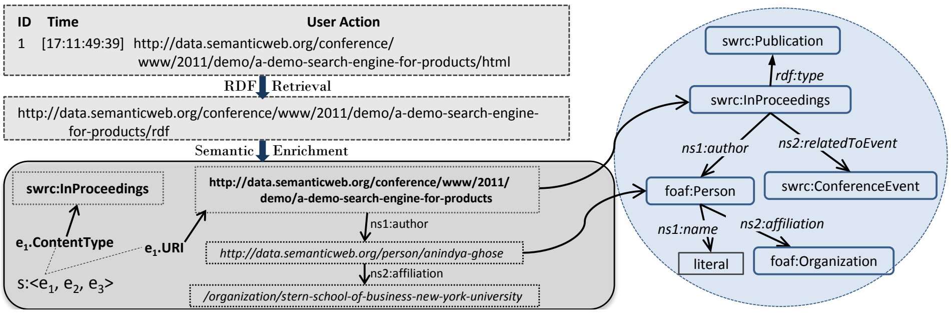

ID
Time
User Action
1
[17:11:49:39]
http://data.semanticweb.org/conference/www/2011/demo/a-demo-search-engine-for-products/html
RDF Retrieval
http://data.semanticweb.org/conference/www/2011/demo/a-demo-search-engine-for-products/rdf
Semantic Enrichment
swrc:InProceedings
http://data.semanticweb.org/conference/www/2011/demo/a-demo-search-engine-for-products
ns1:author
http://data.semanticweb.org/person/anindya-ghose
ns2:affiliation
/organization/stern-school-of-business-new-york-university
e1.ContentType
e1.URI
s:<e1, e2, e3>
swrc:Publication
rdf:type
swrc:InProceedings
ns1:author
ns2:relatedToEvent
foaf:Person
swrc:ConferenceEvent
ns1:name
ns2:affiliation
literal
foaf:Organization

Figure 4.6: Example of semantic enrichment with the Web of Data

Initially, the raw log consists of a syntactic representation of the URL request. Applying the previously described technique, we retrieve the RDF representation of the content behind this URL. We identify the URI of the Web resource that this RDF document describes. Additionally, we identify in the document the domain ontology, in this case SWRC ontology about scientific publications, which we use to enrich the event with additional semantic knowledge.

In this example, we find that the identified resource is a publication of type swrc:InProceedings. We include this class in our knowledge base as subclass of wam:ContentType. Using the RDF data in the retrieved document, we can extend the event context with additional information, such as the proceeding is related to a conference event, the location of the conference, the author of type foaf:Person, the author's affiliation, etc.

[15] http://dbpedia.org/ontology/

4.4.2 Semantic Enrichment using Structured Markup Data

In the previous enrichment technique, we exploited Web sites that provide semantic annotations in pure RDF format with comprehensive domain ontologies. While this technology is increasing in popularity, its deployment is still restricted in the number of sites. For this reason, we also address another category of sites, which provide metadata as structured markup information embedded in the HTML content.

The annotation of HTML elements with structured markup data is a technique increasingly used nowadays by Web publishers to enable search engines, web crawlers, and browsers to extract, automatically process and better understand the content of HTML pages. Several techniques are available to add structured data and augment the visual human-readable information on the Web with machine-processable content. Through simple HTML attributes, Web authors can semantically markup their pages to make them easy understandable for machines, consequently improving their own ranking in search engines.

Recent studies [Bizer et al. 2013, Mika and Potter 2012, Mika 2011] confirm the acceleration of this development, which was particularly boosted in the last two years when prominent applications from Google, Facebook, Yahoo!, and Microsoft have started to deploy the embedded metadata. Major search engines such as Google, Yahoo!, and Bing use the metadata to enrich the search results, therefore increasing the motivation of the Web publishers to markup their pages [16].

These studies have also analyzed the most prevalent semantic markup standards being used today, showing a large-scale and global adoption of RDFa, Microdata, and Microformats. Prevailing standard is Microformats [17], which uses style definitions to annotate HTML text with terms from a pre-defined set of vocabularies. RDFa (Resource Description Framework in Attributes) [Adida and Birbeck 2008] is also a key enabling technology for semantic markup, applied to embed RDF data into HTML pages. Microdata[18] is another widely spread format that is developed recently in the context of HTML5. Semantic annotations in Web pages may use different vocabularies supported by various consumers, for example the Open Graph

[16] http://schema.org/

[17] http://microformats.org/

[18] http://www.w3.org/TR/microdata/

Protocol[19] supported by Facebook, DCMI Terms [20], or schema.org [21] that is currently understood by Bing, Google, Yahoo and Yandex. The continous adoption of these forms of structured data in HTML pages encourages us to exploit such rich corpus of semantic meta-data in the realm of our work.

Algorithm 3: Resource assertions *assertSemResMD*(e_i,b) with Markup Data

```
Input: Event $e_i$ and its corresponding Web domain $b$
Output: ABox assertions $\alpha_r$ for the identified resource and its content types
    //Semantic Resource Identification
 1: RDF representation $\mathcal{D}$=extractRDF($e_i.l$)
 2: Extract resources $\mathcal{R}$=extractResources($e_i.l$, $\mathcal{D}$)
 3: Identify for $e_i$ a mapping resource $R_l$=entityResolution($e_i$, $\mathcal{D}$)
 4: Assertion $\alpha_r$ = { $\mathsf{hasURI}(e_i, R_l)$ }
    //Semantic Types
 5: Identify schemas $\mathcal{S}$= getSchemas($\mathcal{D}$)
 6: For resource $R_l$ find its types $\mathcal{T}_r$=resourceClassification($R_l$, $\mathcal{S}$)
 7: for all $T \in \mathcal{T}_r$ do
 8:     $\alpha_r = \alpha_r \cup \{ T \sqsubseteq \mathsf{ContentType} \}$
 9:     $\alpha_r = \alpha_r \cup \{ \mathsf{contentType}(e_i, T) \}$
10: end for
    //Establish relation of $R_l$ to resources in $\mathcal{R}$
11: for all $r \in \mathcal{R}$ do
12:     $\alpha_{\mathcal{D}}$= getAssertions($R_l, \mathcal{R}, \mathcal{D}$)
13:     $\alpha_r = \alpha_r \cup \alpha_{\mathcal{D}}$
14: end for
```

In Algorithm 3, we describe our approach for exploiting this form of structured data to enrich user browsing behavior models. In order to obtain additional information about the pages that the users have accessed, we extract the domain-level structured objects embedded in their HTML content. This is achieved with specific parsing modules (Alg. 3, line 1), which are able to extract data in RDF representation from a variety of Web documents that are represented in the various formats mentioned above (RDFa, microformats, etc.).

[19] http://ogp.me/

[20] http://dublincore.org/documents/dcmi-terms/

[21] http://schema.org/

In contrast to the previous Web of Data technique where we map the retrieved RDF document to one resource, here the extracted RDF data describe several resources that are not necessarily uniquely identified. As such, we have devised additional methods to locate and uniquely identify resources in the RDF data (Alg. 3, line 3), then accordingly select a representative resource to uniquely describe the browsing event. These methods comprise the *entity resolution* module, which we describe in more details below.

The output of this module is the resource identified with an URI, which we assert in our knowledge base by relating it to the browsing event (Alg. 3, line 4). For this resource, we also find its semantic types expressed in the retrieved RDF representation. These are the classes to which the resource belongs. As in the Web of Data case, we model these classes as subclasses of wam:ContentType, and accordingly establish the relation wam:hasContentType between each of these classes and the browsing event (Alg. 3, line 5-10). Through the entity resolution model we also identify relations among the resources. We then create logical assertions that are added to the knowledge base (Alg. 3, line 11-14).

Entity Resolution. This process, also referred to as record linkage, consists in identifying and linking different manifestations of the same ”real-world entities”. In our approach, an *entity* refers to the *resource* identified in the RDF representation. Resolution is needed because in various domains or multiple data sources the same objects in the real world (e.g. persons, concerts) are referred to and expressed in different ways. For instance, two resources on the same person are described with different name spellings, and annotated with different vocabularies. The goal of the entity resolution module is to resolve resources, first identifying them in the retrieved RDF data, then finding if they represent the same object in real world. Following a Linked Data principle [Bizer et al. 2009], we express these links in RDF using owl:sameAs statements.

Different Web sites generally use different schemas to annotate their HTML elements. Therefore, we apply schema mapping by a set of rules that map predicates, which belong to different schemas, but have the same semantics (e.g. predicate dc:title [22] and predicate og:title [23]).

In the next step, we apply a scoring function to find how closely the attributes of two resources match. We automatically group resources based on their type. Afterwards, we compare the resources of the same type based on the values of the attributes, after having aligned them by the previously mapped predicates. The values of the attributes are compared using the Levenshtein distance, which is a string metric for measuring the difference between two sequences.

The Levenshtein distance between two strings is defined as the minimum number of edits needed to transform one string into the other, with the allowable edit operations being insertion, deletion, or substitution of a single character. In our case, we use a threshold value of the edit distance between the attribute values. For instance, two resources of type $Conference$ will be linked with the relation rdf:samesAs, if they have the same value of the attributes $location$ and $time$, whereas the values of their $title$ attributes have an edit distance smaller than the defined threshold. We tuned the threshold value on a sample of previously matched resources.

We illustrate the semantic enrichment process with an example in Figure 4.7. We consider a session of three browsing events. For each of these events we show the respective URL. For the semantic enrichment process, we start with each URL independently, read the content of the HTML document under this URL, and then extract the RDF structured data embedded in it. For instance, for the first URL at *eventbrite.com*[24], we identify resource r_1 and extract its metadata, such as the followings:

- og:type is eventbriteog:event
- og:title is "Kings of Leon Concert"
- ical:dtstart[25] is "Tuesday, February 18, 2014 at 7:30 PM (EST)"

Resource r_1 is further related (via ical:location) to a resource of type vcard:Address[26], which has the metadata:

- vcard:street − address is "555 Borror Drive"
- vcard:locality is Columbus"

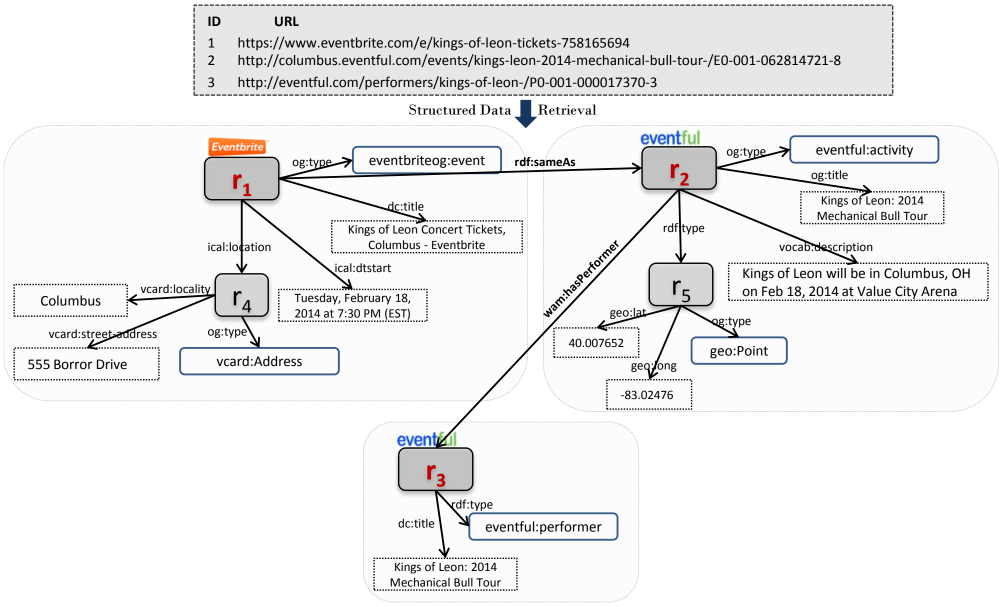
[22] dc: <http://purl.org/dc/terms/>

[23] og: <http://opengraphprotocol.org/schema/>

[24] https://www.eventbrite.com/e/kings-of-leon-tickets-758165694

[25] ical: <http://www.w3.org/2002/12/cal/icaltzd#>

[26] vcard: <http://www.w3.org/2006/vcard/ns#>



ID	URL
1	https://www.eventbrite.com/e/kings-of-leon-tickets-758165694
2	http://columbus.eventful.com/events/kings-leon-2014-mechanical-bull-tour-/E0-001-062814721-8
3	http://eventful.com/performers/kings-of-leon-/P0-001-000017370-3

Figure 4.7: Example of semantic enrichment with structured markup data

The publisher of this Web page has used different schemas for the semantic annotation. We find additional metadata used for the annotation (not shown in Figure 4.7):

- dc:title is “Kings of Leon Concert Tickets, Columbus - Eventbrite” [27]
- og:locality is “Columbus”
- og:postal − code is “43210”
- og:street − address is “Schottenstein Center 555 Borror Drive”

For the second URL at *eventful.com*[28], we identify the resource r_2 with the following metadata:

- og:title is “Kings of Leon: 2014 Mechanical Bull Tour”
- og:type is “activity”
- vocab:description is ”Kings of Leon will be in Columbus, OH on Feb 18, 2014 at Value City Arena.”

[27] dc:<<http://purl.org/dc/terms/>

[28] http://columbus.eventful.com/events/kings-leon-2014-mechanical-bull-tour-/E0-001-062814721-8

In the extracted metadata, we find that r_2 is related to another resource of type geo:Point with the following attributes:

- geo:lat[29] is “40.007652”
- geo:long is “-83.02476”, which corresponds to the locality Columbus in Ohio, USA

The third URL[30] is mapped to resource r_3 whose rdf:type is eventful:performer and its dc:title is “Kings of Leon”.

We perform entity resolution and compare the respective metadata of the resources, at first globally mapping the predicates (e.g. dc:title and og:title), then measuring string distance between the values of the predicates. As such, we establish owl:sameAs relation between the identified resources, e.g. concluding that resources r_1 and r_2 represent the same real world events. The wam:hasPerformer is inferred between the resources r_2 and r_3, because resource r_3 is of type eventful:performer and its description is found in the RDF representation of the resource r_2.

[29] geo:<http://www.w3.org/2003/01/geo/wgs84_pos#>

[30] http://eventful.com/performers/kings-of-leon-/P0-001-000017370-3

4.5 Semantic Enrichment using Supervised Learning

After the deployment of the formalization and automatic semantic enrichment approach, we generate a session $s_k = \langle e_1, e_2, ..., e_n \rangle$ of semantically-annotated browsing events. For some of the events, we are able to automatically find and assign a `ContentType` class. Yet, there are also events in s_k for which no `contentType` class could be retrieved. Hence, we follow a second step of semantic enrichment that comprises a supervised technique for learning the class, based on the observed examples (i.e. already formalized events in the overall sessions).
Finding the `contentType` class of a browsing event can be formulated as a classification problem, borrowing from the field of machine learning. The task is to assign a particular event to a predefined class, being in our case the `ContentType` of this event. Hence, we formulate this task as a classification problem, in which we have to learn a function $f : E \rightarrow \mathcal{C}$ that maps an event $e_i \in E$ s.t. $e_i = (l_i, \mathcal{T}_i, P_i, t_i)$ (as in Def. 6) to an output class $c_i \in \mathcal{C}$. In our case, $\mathcal{C}$ is a set of classes belonging to an ontology $\mathcal{O}$.

4.5.1 Learning Semantic Types of Resources

In our approach, we use the generalized formulation of multi-class SVM learning [TSOCHANTARIDIS et al. 2004]. We are interested on the problem of learning a function $f : \mathcal{X} \rightarrow \mathcal{Y}$, which maps input instances $\mathbf{x} \in \mathcal{X}$, which in our setting consist of the formalized events, to discrete outputs $\mathbf{y} \in \mathcal{Y}$ that consist of arbitrarily numbered labels representing `ContentType` classes in our ontology.
Let's consider the case of finding a function f that maps each event $\mathbf{x}_i$ from usage logs to one of the classes in $\mathcal{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_n\}$. The task is to learn a discriminant function $F : \mathcal{X} \times \mathcal{Y} \rightarrow \Re$ over input/output pairs, so that for a given input $\mathbf{x}$, we can make a prediction by maximizing F over the response variables:

$$F(\mathbf{x}; \mathbf{w}) = \underset{\mathbf{y} \in \mathcal{Y}}{argmax}\ F(\mathbf{x}, \mathbf{y}; \mathbf{w})) \tag{4.44}$$

In our case, we deal with a multi-class classification problem [CRAMMER et al. 2001], where $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_K\}$ is the input set of events in the log sessions, $\mathcal{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_N\}$ is the set of output classes from ontology $\mathcal{O}$, and $\mathbf{w} = (w_1, ..., w_N)$ is a stack of vectors with w_n being a weight vector for the class $\mathbf{y}_n$. We use the following formulations of the linear discriminant functions F:

$$F(\mathbf{x}, \mathbf{y}_n; \mathbf{w}) = \langle \mathbf{w}_n, \Phi(\mathbf{x}) \rangle \tag{4.45}$$

where $\Phi(\mathbf{x}) \in \Re$ is the vector of numeric features extracted from $\mathbf{x}$.
SVMs, then, solve the following optimization problem:

$$\underset{\mathbf{x},\xi}{min} \frac{1}{2} \sum_{i=1}^{N} || \mathbf{w}_i ||^2 + \frac{C}{K} \sum_{i=1}^{K} \xi_i \tag{4.46}$$

$$\forall i, \forall \mathbf{y} \in \mathcal{Y} \setminus \mathbf{y}_i : \langle \mathbf{w}, \Phi(x_i) \rangle \geq 100\Delta(\mathbf{y}_n, \hat{\mathbf{y}}_n) - \xi_i \tag{4.47}$$

with regularization parameter C and slack variables ξ_i for margin violations [JOACHIMS et al. 2009].
The learning algorithm optimizes the error rate during training, minimizing prediction loss defined by a function $\Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \Re^D$, where $\Delta(\mathbf{y}_n, \hat{\mathbf{y}}_n)$ is the loss of predicting $\mathbf{y}_n$ when the correct output is $\hat{\mathbf{y}}_n$. The loss function, which quantifies the mismatch between the predicted class and the expected correct output, in our case returns 0 if $\mathbf{y}_n$ equals $\hat{\mathbf{y}}_n$, and 1 otherwise.
A crucial part of the classification approach is engineering the features for the numeric vector representation of the input instances, i.e. defining the function $\Phi(\mathbf{x})$ in Equation 4.44. We have designed different categories of features, which we explain in details in Section 4.5.2.

Reasons for choosing structural SVMs

There are several reasons for choosing structural SVMs as our classification method. Firstly, SVMs in general are shown to perform better in building complex and accurate models [JOACHIMS et al. 2009], particularly in settings similar to ours such as Web page categorization or purely URL-based page classification [KAN and THI 2005, BAYKAN et al. 2011]. Secondly, SVMs deal very well with sparse and highly dimensional data, as is the case of the huge and heterogeneous amounts of cross-site usage logs, which lead to feature vectors that are large and highly sparse.

At last, structural SVMs enable learning for complex and interdependent objects of the output space, forming the basis for further extensions of our approach to learn a formal, structured ontology with class relationships for the classification of events in the usage logs.

4.5.2 URL-based Feature Categories

We design different categories of features for the classification of event types. We explore features that we believe are applicable to a broad range of URLs and likely to positivly impact classification. The goal is to extract as much information as possible from the URL, as the sole source of raw information explicit in the user browsing logs.

We first generate whole tokens from the URLs and letter n-grams of tokens [Kan and Thi 2005]. We also engineer *sequential features*, such as sequences of pairs of tokens in the URL, referred as the *Precedence Bigrams*. We further propose a new feature category (*Sequential Neighbors*) based not only on the URL of the event, but on the sequential information related to the session in which the event belongs. In this case, the tokens of the neighboring events are also included as features. In Table 4.1, we list the feature categories and illustrate them with an example.

Sample URL	http://data.semanticweb.org/web/anindya-ghose

Feature category (tag)	**Example**
Token (T)	data semanticweb org web anindya ghose
Sequential Trigrams (N)	dat|ata|org|sem|ema|man|ant|nti|tic|icw|cwe|web|org|web|ani|nin|ind|ndy|dya|gho|hos|ose
Precedence Bigram (P)	data>semanticweb data>org data>web data>anindya data>ghose semanticweb>org ... web>anindya web>ghose anindya>ghose
Sequential Neighbors (S)	data semanticweb org conference www 2011 demo search engine for products eventbrite event 8899503655 webcong2012

Table 4.1: URL feature categories and examples

Tokens as Features. In order to produce features of this category, we preprocess the URL by transforming it into a lower-case form, then splitting it into strings of letters (tokens) applying as delimiters punctuation marks, numbers, or other non-letter characters. No stemming is performed. Tokens of length less than 2 are filtered out. In Table 4.1, we show an example of the tokens generated for the given sample URL.

Sequential Trigrams as Features. For this feature category, we split the URL in the same tokens as explained above, then generate letter trigrams from the tokens, which are sequences of exactly 3 letters. Shorter tokens are left unchanged. An important advantage of n-grams (in our case trigrams) over tokens is the potential to detect sub-words (e.g. "*web*" within "*semanticweb*"), without demanding an explicit list of valid terms.

Precedence Bigram Features. The sequence of tokens in the URL is also an important aspect that capture additional latent information. For example, considering the case of a URL with the structure "*/states/france/cities/lyon*" and another URL of the form "*/france/lyon*". They are composed of following token sequences: *"states france cities lyon"* and *"france lyon"*. The features that are based solely on tokens of strings fail to capture the similarity between sequences, because the token *cities* plays a crucial precedence relationship. We can capture this latent information by a feature category that models left-to-right precedence between tokens: $europe > germany$. As shown in Table 4.1, a feature example for the sample URL would be: $anindya > ghose$.

We generate precedence bigram features as sequences of token pairs. An example is shown in Table 4.1, where we illustrate the pairs of tokens for the sample URL modeling left-to-right precedence.

Sequential URL Neighbor Features We further propose a new feature based not only on the information in the URL of the event, but on the sequential information related to the session in which the event belong. In this case, the tokens of the neighboring events are also included as features. For instance, as sequential neighbor features for the sample URL in Table 4.1, we add the tokens of the URL[31] belonging to the event preceding it in the session logs (Figure 4.2).

4.5.3 Classification Procedure

For our classification task, we apply the following procedure (as in Algorithm 4):

Algorithm 4: Procedure for supervised classification of events

```
1: Data preprocessing
2: Feature engineering
3: Model selection
4: Training with default parameters
5: Cross-validation to find the best regularization parameters
6: Testing and evaluation
```

Data Preprocessing. It comprises a series of steps for the transformation of browsing logs to a format appropriate for training and testing. After the logs have been semantically formalized using our formalization approach, we select a portion of the data for the classification problem.

Initially, since the formalized logs are represented as RDF triples and stored in a repository, using SPARQL queries we extract two sets of data for training and testing, each of them containing a session ID, the URL of event and the order of the event belonging to that session. We then prepare training and test datasets, respectively. Since supervised learning needs labeled data, a part of those are generated from the mapping to the domain ontology, which serve as ground truth values. The labels[32] that are not found in the ontology are annotated manually.

Feature Engineering. To apply a machine learning algorithm, URLs have to be mapped to numerical feature vectors. Each instance (URL) in the input space is represented as a vector of real numbers. In order to construct such feature vectors, we follow a series of processing steps aligned with our definition of the features. This process includes tokenization, n-gram generation, precedence bigrams and

[31] `http://data.semanticweb.org/conference/www/2012/demo/a-demo-search-engine-for-products`

[32] Terms *label* and *class*, as well as *instance* and *event* are used interchangeably.

sequential neighbor features formation. Tokens or ngrams derived from the URL of the event serve as binary features.

Model Selection. We apply the linear kernel of structural SVMs, motivated by the following reasons: high dimensionality of the feature vectors, huge number of features, and high number of classes/labels. In addition to varying the number of features, SVM performance is governed by the parameter C (the penalty imposed on training examples that fall on the wrong side of the decision boundary). We apply the model with different values of the regularization parameter C.

Training and Validation We have conducted evaluation experiments using datasets of real-world usage logs. In section 4.6, we provide details on the characteristics of the datasets used for training and testing. We report on the evaluation results of these experiments.

4.6 Experimental Results

We have performed various experiments in order to prove the feasibility and effectiveness of our formalization and semantic enrichment techniques. We have used datasets of user logs from different Web sites with the goal of showing the feasibility of our approach when dealing with heterogeneous content. In the following sections, we report on the methodology used to evaluate each of the enrichment techniques and the results of our experiments.

4.6.1 Enrichment with the Web of Data

Implementation. We provide a Java SE implementation of the introduced formalization approach, deploying the steps of processing usage logs, cleaning, and formalization with WAM ontology. The consistency of the ontology is checked with Pellet 1.5.2 reasoner. We further implemented the step of semantic enrichment of events with predefined ontologies from the Web of Data domains. We use the APIs of Apache Jena [33], an open source Semantic Web Framework, in order to read Resource Description Framework (RDF) graphs, serialise our RDF triples using the popular RDF/XML format, as well as to update and query our triple store of semantic models.

The formalized sessions and browsing events are serialized in RDF representations, which are afterwards imported via OpenRDF Sesame Core 2.6.0 API [34] into a Sesame Framework [35] repository.

Datasets. In order to show the feasibility of our semantic enrichment approach, we performed several experiments in which we have processed and semantically formalized logs extracted from the USEWOD dataset [Berendt et al. 2012]. The USEWOD dataset consists of Common Logs Formal (CLF) [36] server logs from major web servers publishing information as Linked Data. In particular, we worked with logs from the following Linked Open Data sites:

[33] http://incubator.apache.org/jena/

[34] http://www.openrdf.org/doc/sesame2/api/

[35] http://www.openrdf.org/

[36] http://httpd.apache.org/docs/current/logs.html

- DBpedia[37]: slices of server log data from one of the central repositories in the Web of Data. DBPedia contains structured content from the information created in Wikipedia. The logs of this dataset spann across several months. For our experiments, we extracted logs of the time period 01/07/2009–12/07/2009 (see Table 4.2).
- SWDF [38]: Semantic Web Dog Food (SWDF) is a constantly growing dataset of publications, people and organisations in the Web and Semantic Web area, covering several of the major conferences and workshops. The overall set of SWDF logs contain two years of requests to the server from about 12/2008 until 12/2010. For our experiments, we extracted a part of these logs of the time period 01/07/2009–12/07/2009 (see Table 4.2).

The absence of publicly available cross-site usage logs from the Web of data is a strong reason for choosing the USEWOD datasets in our experiments. Despite the fact that these data are gathered independently from the two Web servers, we design our techniques with the aim of discovering patterns of cross-site behavior, investigating traces of users navigation from one Web domain (e.g. DBpedia) to another (e.g. SWDF, various search engines, etc.).

	SWDF	DBPedia
Monitoring Period	01/07/2009–12/07/2009	01/07/2009–12/07/2009
Nr. sessions	2831	31893
Avg. nr. daily sessions	235.92	2899
Mode nr. events/session	4	10
Nr. events	10437	>426000
Nr. triples	277788	$>$ 3million
% events with $contentType$	83%	81%

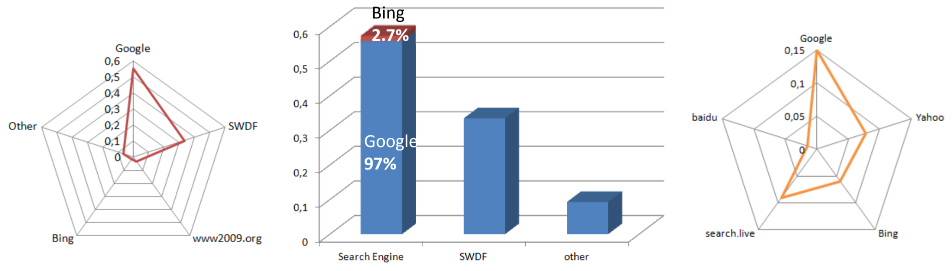
Table 4.2: Results of semantic enrichment with Web of Data

Experimental Results. We show the results of our experiments in Table 4.2. For the SWDF dataset, we are able to formalize 2831 sessions in the selected monitor-

[37] http://dbpedia.org

[38] http://data.semanticweb.org

ing period. For the same time period, there are 31893 formalized sessions from the DBpedia dataset. DBPedia, being a large provider of structured data, contributes with nearly 426 thousand browsing events, leading to more than 3 million RDF triples. We applied the enrichment technique in order to retrieve the semantic content type for each event. We are able to find the content type class of 83% of events in SWDF dataset and 81% of events in DBPedia.



(a) SWDF

(b) SWDF

(c) Dbpedia

Figure 4.8: Distribution of daily sessions initiated at particular domains: (a) The majority of the daily sessions (57.2%) in SWDF dataset start with a search at Google (.com,.de, etc.) engine. In 33.5% of the sessions users are directly accessing SWDF; (b) From the daily sessions in SWDF dataset that are initiated in search engines, an average of 97% start at Google and only 2.7% at Bing; (c) The majority of the daily sessions (15%) in DBpedia dataset start from Google search engine, less than 1% start at other search engines (e.g. Yahoo or Bing).

We perfom an analysis to discover other characteristics in these datasets. The results of the analysis are graphically illustrated in Figure 4.8. We observe that SWDF has between 161 and 309 user sessions daily, with an average of 235.9 sessions/day. The majority of the daily sessions (57.2%) start at Google search engine and then continue at SWDF, while 33.5% start with the user directly accessing SWDF. From the daily sessions initiated in search engines, an average of 97% start at Google (.com,.de, etc.) and only 2.7% at Bing.

There are very few sessions from human users containing SPARQL queries (in average 1.46% daily sessions). The majority of sessions containing SPARQL queries belong to machines/bots, which we have filtered out with our formalization approach.

Through the implementation and conducted experiments, we practically prove that the formalization algorithm is effective for different domains and the enrichment approach is feasible. Overall, we processed nearly one month of usage logs from large Web sites (such as DBpedia), demonstrating that the approach is viable. Furthermore, we observe that through the proposed enrichment technique we are able to retrieve the semantic content type of more than 80% of browsing events in the logs.

4.6.2 Enrichment with Structured Markup Data

Implementation. We provide a proprietary implementation of all the modules of the enrichment technique with structured markup data described in Section 4.4.2. An external library used in our approach is the metadata parser Apache Anything To Triples (any23)[39]. It is provided as library, web service and command line tool to extract from Web documents structured data in RDF format. It supports various input formats such as RDF/XML, Turtle, RDFa, Microformats, and HTML5 Microdata.

For our experiments, we prepared new real-world datasets that combine user browsing logs at different Web sites with semantic metadata embedded in the Web pages.

Toolbar Logs Dataset. The dataset of user logs (Table 4.3) consists of browsing behavior data registered via the Yahoo! Toolbar, which tracks HTTP requests upon agreement of users that have installed this toolbar. From a huge dataset, we extracted a sample of user logs recorded in a timeframe of five weeks, consisting of 526 sessions of 494 users. We identified 1683 unique URLs in these logs. The minimum length of the session is 2, which means the session consists of two browsing events (or that the user visited two URLs in the sites of interest). The average length of the sessions is 4.132, whereas the mode of session length is 3. Sessions of length 3 comprise 38.86% of all sessions in this dataset.

[39] `https://any23.apache.org/`

	Yahoo! Toolbar Dataset
Logs period	01.Jul.2012 - 07.Aug.2012
Nr. users	494
Nr. log entries	2244
Nr. unique URLs	1683
Nr. user sessions	526
Average session length	4.132 (10.31% sessions)
Mode session length	3 (38.86% sessions)
Min session length	2 (15.65% sessions)
Max session length	15 (0.18% sessions)
Nr. cross-site sessions	14.0 (2.58% sessions)

Table 4.3: Statistics of the toolbar logs dataset

Experimental Results. The result of the semantic enrichment approach is a dataset consisting of browsing events leveraged with the metadata extracted from the Web pages accessed in the logs. Statistics on this dataset are illustrated in Table 4.4.

Our approach is open-domain, but to enable a more detailed evaluation of the enrichment results, we focused on the user logs tracked at three Web sites: Eventful [40], eventbrite [41], and Upcoming[42]. These are sites of event (concert, conference) advertisement nature. We filter all the sessions that contain at least one URL from these sites. After duplicate detection, we filter all the unique URLs, so that each represents a resource URI from the respective site. Overall, we find that half of the resources in the dataset are from Eventful, whereas 18.78% from Upcoming and 30.84% from Eventbrite.

We observe that Eventful provides a large amount of structured description in their pages. We are able to find 76.44% of resources annotated with at least a type. At Eventful, the most used predicate for such annotation is schema.org:type with object rdf:Event [43].

[40] http://eventful.com

[41] http://eventbrite.com

[42] http://upcoming.yahoo.com

[43] <http://schema.org/Event>

	D_1	D_2	D_3
PLD (sites)	Eventful	Upcoming	Eventbrite
Nr. Resource URIs	6939	2586	4247
Resource URIs per site	50.38%	18.78%	30.84%
Typed Resources	76.44%	82.29%	81.21%
Resources of type *activity*	22.08%	28.05%	80.60%
Resources of type *search*	3.08%	53.63%	8.62%
Total resource URIs	13772		
Total resource URIs with type	79%		
Formats	RDFa Microdata Microformats		
Top namespaces	http://schema.org/ http://opengraphprotocol.org/schema/ http://purl.org/dc/terms/ http://www.w3.org/1999/xhtml/vocab# http://www.w3.org/1999/02/22-rdf-syntax-ns# http://www.w3.org/2002/12/cal/icaltzd# http://www.w3.org/2006/vcard/ns# http://www.w3.org/2003/01/geo/wgs84_pos#		
Top predicates	http://www.w3.org/1999/02/22-rdf-syntax-ns#type http://opengraphprotocol.org/schema/type http://purl.org/dc/terms/title http://opengraphprotocol.org/schema/title		

Table 4.4: Statistics of the dataset resulting from the semantic enrichment approach with structured markup data

The site Upcoming offers rich semantic annotations of their respective pages. We found that 82.29% of resources have a type, which is most often (28% of the cases) annotated with ical.org:vevent[44] of the microdata format. There are not so many resources of type activity being browsed by users, but rather more pages that we classify to be of ”search” type. These are pages under URL of the form *http://upcoming.yahoo.com/search/?q=concerts&loc=Lyon*, denoting a search query within the site. For these pages, which comprise 53.63% of the unique URLs browsed at Upcoming, there are also rich structured markup data provided in the HTML code. The pages resulting from search invocation are 69.86% of the time annotated with typed resources.
At Eventbrite we are able to identify nearly 81% typed resources. The majority of these resources, more precisely 80% of them, are annotated with the class eventbriteog.org:event and predicate og:type from the Open Graph protocol[45]. Overall, we are able to identify 79% typed resources for the whole dataset of logs. An interesting observation with respect to user browsing behavior in these sites is that at Upcoming users tend to navigate more by performing search, whose resulting pages are very-well annotated by the content provider. At Eventful there are very few invocations of search URLs, with more requests to pages that have clean annotation with activity-related metadata. At Eventbrite there are also very few search urls being invoked (8.62%), but less than half of the pages resulting from search are annotated with some form of metadata.
Furthermore, we observe that these sites apply all the different markup formats for semantic annotation, i.e. RDFA, OGP, microdata, microformats. We examine that even within a page various formats and schemas are used to annotate the same attributes. Among the most popular namespaces are schema.org, OGP, and DCMI Terms.

[44] http://www.w3.org/2002/12/cal/icaltzd#
[45] http://opengraphprotocol.org

4.6.3 Enrichment via Supervised Learning

We performed another set of experiments to evaluate the effectiveness of our technique used for learning semantic resource types in a supervised way. In this section, we report on the experimental setup and evaluation results that demonstrate the efficacy of our approach.

Datasets. For our supervised learning experiments, we used two datasets D_1 and D_2 of different sizes extracted from the repository of events generated from the formalization and enrichment approach. These are the events belonging to two weeks of the SWDF dataset shown in Table 4.2. For both datasets we prepared training and testing sets. The test sets contain events for which the content type was not automatically found via enrichment. We report on the characteristics of these datasets in Table 4.5.

	Dataset D_1		Dataset D_2	
	Training set	**Test set**	**Training set**	**Test set**
Nr. events/set	974	1152	4676	4957
Total nr. events	2126		9633	
Nr. classes	66		82	

Table 4.5: Training and testing datasets used for the evaluation of semantic type learning technique

For dataset D_1, we selected usage logs of two random consecutive days, extracting the events of one day (3. July) for the training and events of another day (2. July) for testing. Whereas for D_2, we chose a larger set comprising the logs of all the days from both weeks.

Methodology and Evaluation Metrics. We used the implementation $structSVM$[46] of structural SVMs with the multi-class formulation. After experimenting with different values of the regularization parameter C, we selected the value 5000 to be the best one. For training, we follow a three-fold cross-validation approach.

[46] http://svmlight.joachims.org/svm_struct.html

To evaluate the performance of our classification approach, we use the F-measure metric as the harmonic mean of precision (π) and recall (ρ). It is calculated as follows:

$$\pi_i = \frac{TP_i}{TP_i + FP_i} \tag{4.48}$$

$$\rho_i = \frac{TP_i}{TP_i + FN_i} \tag{4.49}$$

$$F_i = \frac{2\pi_i\rho_i}{\pi_i + \rho_i} \tag{4.50}$$

where TP_i (True Positives) is the number of instances assigned correctly to class i; FP_i (False Positives) is the number of instances that do not belong to class i, but are assigned to class i incorrectly; and FN_i (False Negatives) is the number of instances not assigned to class i, but which actually belong to this class. The F-measure values calculated for each class with Equation 4.50 are in the interval (0,1), such that larger values correspond to higher classification quality.

To compute the overall F-measure score of our multi-class classification problem, we use *macro-averaging* (Equation 4.51) as a binary evaluation measure across the overall N classes:

$$macroF_1 = \frac{\sum_{i=1}^{N} F_i}{N} \tag{4.51}$$

Macro-averaging is a well-known metric generally used to calculate binary evaluation measures across several classes.

Experimental Results. In our experiments, we use the token feature as the baseline. We report on the zero/one-error (percentage of misclassified instances) and macro-F1 measures of our results as illustrated in Table 4.6. Since we are interested to see how precise is the approach in mapping a URL to the correct class, we pay particular attention to the zero/one-error metric. As can be observed in the table, the *trigram* feature (N) and *sequential neighbor* features (S) particularly play an important role in the classification accuracy. The error decreases for these

	Dataset D_1			Dataset D_2		
Feature Category	Nr. Features	zero/one-error	Macro-F1	Nr. Features	zero/one-error	macro-F1
Token (T)	1357	14.40%	0.79	4341	13.04%	**0.75**
Trigram (N)	4673	14.41%	**0.84**	11205	12.99%	0.69
Precedence Bigram (P)	3385	14.75%	0.82	11060	**11.80%**	0.58
Sequential Neighbors (S)	4071	13.54%	0.67	13023	**12.02%**	**0.63**
S+P	6099	14.06%	0.73	15647	N/A	N/A
N+S	7387	**13.45%**	**0.74**	19887	N/A	N/A

Table 4.6: Macro-F1 and zero/one error of the experimental results with regularization parameter $C = 5000$

features, when compared to the results obtained from the baseline with Token (T) only (14.40% for dataset D_1 and 13.04% for dataset D_2).

Furthermore, for D_1 we observe that the combination of features N and S yields the best results, since the error is the smallest, while still keeping a high value of the macro-F_1 measure. For D_2 we note that the *precedence bigram* feature (P) gives the best classification results in terms of the zero/one-error rate[47]. Still, as in D_1, the impact of the *sequential neighbor* feature yields the best combination of the lowest error and overall high averaged F_1 score. This proves our expectation that users sequentially browse related resources, which can help us to derive missing semantic types.

4.7 Related Work

The works related to ours may be grouped as follows:

Browsing behavior modeling at multiple sites. Interest to characterize online behavior has started much earlier with works such as those of Catledge *et al.* [CATLEDGE and PITKOW 1995], and Montgomery *et al.* [MONTGOMERY and FALOUTSOS 2001] that try to identify browsing strategies and patterns in the Web. Browsing activity has been studied and modeled by Bucklin *et al.* [BUCKLIN and SISMEIRO 2003] and others who usually exploit the server-side logs of visitors in a single Web site.

With respect to modeling the browsing behavior at multiple Web sites, Downey *et al.* [DOWNEY et al. 2007] propose a state machine representation for describing search activities. In a later work [DOWNEY et al. 2008], the authors deliver a study of browsing behavior after the user departs the search engine and begins to follow an information thread through the Web. Park and Fader [PARK and FADER 2004] present a stochastic timing model of cross-site user visit behavior, using information from one site to explain the behavior at another. Johnson et al. [JOHNSON et al. 2004] study online search and browsing behavior

[47] The experiments on D_2, whose results are reported as N/A , were not supported by our machine because of the high dimesionality of feature vectors.

across competing e-commerce sites. The works in this category do not particularly apply semantic techniques or ontologies for behavior modeling.

There also exists a multitude of works that investigate modeling of search behavior. We are excluding the references to these approaches, since our work focuses on general Web browsing behavior and not particularly search query modeling.

Semantic formalization of usage logs. This group of works include approaches that aim at capturing events of user interactions with annotated Web pages and lifting these events to RDF e.g. Stühmer *et al.* [STÜHMER et al. 2009]. Tvarozek *et al.* [TVAROŽEK et al. 2007] present personalization techniques that exploit annotations of user characteristics.

The UCIAD platform[48] also deploys annotation of user-centric activity data, relying on pre-defined patterns to characterize resources, as part of the platform setup [D'AQUIN et al. 2011]. They propose an upper ontology to represent traces of activities performed by agents on particular Web pages. This is referred to as the Trace Ontology and is modeled with OWL[49]. It captures information related to HTTP server logs.

The task of semantically enriching the items accessed by Web visitors has also been addressed by Mobasher *et al.* [MOBASHER et al. 2003]. In this work, they make use of domain-specific wrappers with pre-defined mining rules to extract class and attribute instances from Web sites. In this approach, a pre-specified reference ontology is engineered and used for each site.

Ontologies in usage mining. There is an extensive body of work dealing with usage log analysis and mining, but we focus on the combination of these techniques with semantic technologies, which start with contributions such as Stumme *et al.* [STUMME et al. 2002] and Oberle *et al.* [OBERLE et al. 2003]. In this field, research has been mostly focused on search query logs or user profiling. Other approaches that exploit semantics for extracting behavior patterns from Web navigation logs are presented by Yilmaz *et al.* [YILMAZ and SENKUL 2010] and Mabroukeh *et al.* [MABROUKEH and EZEIFE 2009]. While Yilmaz com-

[48] http://uciad.info/ub/

[49] https://github.com/uciad/UCIAD-Ontologies/blob/master/trace.owl

bines ontology and sequence information for sequence clustering, Mabroukeh investigates sequential pattern mining and next step prediction. Vanzin *et al.* [VANZIN et al. 2005] present ontology-based filtering mechanisms for the retrieval of Web usage patterns. The work of Adda *et al.* [ADDA et al. 2010] tackles the problem of mining meaniningful usage patterns and exploit the impact of ontologies to solve this problem. These works are restricted to only one domain and do not handle cross-domain browsing behavior. Furthermore, the majority of the works assume the availability of annotated Web pages. The mapping of URLs to concepts in an ontology is often done manually. While for a single domain this can be affordable, in an open setting with various domains the manual approach is infeasible.

The difference of our work with this large body of contributions lays particularly in our focus to address in depth the problem of usage data formalization, while offering a core ontology to structure user browsing logs and automatic techniques for their semantic enrichment.

URL-based Web page classification. The task of Web page classification into topics using only the information contained in the URLs has been investigated in several works [BAYKAN et al. 2011, BAYKAN et al. 2009, KAN and THI 2005, ABRAMSON and AHA 2012, KOPPULA et al. 2010]. There are various reasons that motivate the deployment of solely URL-based features for classification, such as the need to provide very fast methods, perform content filtering before the download of a page, pre-identify the page language, or infer the topic of a page before the download for focused crawling.

Kan *et al.* [KAN and THI 2005] perform Web page classification using URLs alone, showing that this method is magnitudes faster than typical classification approaches that fetch and analyze entire pages. In this work, there are various features generated from the URL, including tokens, sequential features, and orthographic features that capture salient patterns in the URL. They introduce features such as position, length, and sequence of tokens. Through their experiments, it is demonstrated that URL-based methods in some cases outperform full-text and link-based classification approaches.

Baykan *et al.* [BAYKAN et al. 2009] present an approach for Web page classification into different topics such as news, sports, shopping, adults, etc. for pages in English language. The features are engineered solely on tokens and n-grams found in the URLs. They perform training of separate binary classifiers for each topic. Their experiments show that SVM performs better than other algorithms, and page short summaries (snippets) together with the URL features lead to a considerable improvement of the F-measure.

In a follow-up work [BAYKAN et al. 2011], the authors provide a comprehensive study of features and algorithms for URL-based topic classification. This work describes the various methods applied to map URLs to Features, and four machine learning algorithms used also in related work and shown to outperform other learning techniques. Through extensive experiments the study shows among other findings that features have more impact than the classification algorithm on the performance, boosting methods that combine different algorithms can help to improve classification accuracy, and it is particularly challenging to perform classification when URLs have unseen tokens or when there is inconsistency of topic definitions among datasets.

4.8 Summary

In this chapter, we addressed the task of modeling user browsing behavior in an open Web setting.We presented an approach for the formalization of user Web behavior based on a novel Web browsing Activity Model (WAM). A crucial part of the formalization is a two-staged semantic enrichment of logs, which maps them to events with comprehensible content types from the application domain. We initially presented automatic techniques to leverage logs with semantic descriptions extracted from existing structured knowledge offered in the Web sites. We exploited the structured descriptions of information resources in the Web of Data and in the form of embeddings within HTML pages. These methods are suited for an open Web setting and do not rely on domain-dependent heuristics or a centralized knowledge base.

To annotate the remaining browsing events tracked in sites that do not provide formal domain ontologies or other forms of structured markup metadata, we deployed a supervised learning technique to infer the content type of browsing events. For this technique, we introduced a multi-class classification formulation of the problem. We explored for the first time the use of Support Vector Machines with structural and interdependent output spaces, as well as engineered new URL-based sequential features for the classification of resources with missing content type.

The semantically-leveraged logs provide an added-value in comparison to their syntactic representation in various ways: allow for more expressive formulation of queries to discover user navigation patterns; serve as useful input for techniques, such as semantic pattern mining, next-step navigation prediction or user clustering, which usually assume that these semantics of logs exists or are manually derived. Another benefit is the potential to extend these techniques to deal with cross-site browsing data and not only data restricted to a single Web site.

The semantic formalization of user browsing behavior lays the foundation for effective techniques of behavior pattern analysis. Additional dynamic aspects of user browsing behavior can be discovered if we enable reasoning not solely with semantic constraints, but also with temporal conditions. For this purpose, as an extension of our formalization approach, we introduce in Appendix B.1 a framework for querying expressive patterns of user browsing behavior. We present a novel formalism to express queries using a temporalized description logic called $\mathcal{DL}$-LTL , which combines $\mathcal{SHOIN}(\mathbf{D})$ with Lineal Temporal Logic. Alongside the formalism, we provide a query answering mechanism, which is based on a model checking technique. This allows to automatically retrieve sessions of user browsing events that satisfy a set of semantic and temporal conditions.

We implemented the proposed formalization approach and the two-stage semantic enrichment approach, and performed experiments with real-world datasets of logs.. To validate the feasibility of our approach, we processed over 30 thousand user browsing sessions from datasets of logs collected from Linked Open Data servers and Yahoo! Toolbar .

Through our enrichment techniques we were able to leverage logs with additional structured descriptions, after having mapped a log entry to a typed resource from the respective Web domain. When exploiting pages from Web of Data we were successful in identifying nearly 80% of such typed resources. Whereas, in the enrichment approach with HTML markup data the proportion of identified typed resources varies between 76%-82% for the targeted three Web sites. Furthermore, we experimentally verified that the extension with the supervised classification technique increases the annotation accuracy.

CHAPTER 5

Collective Cross-domain Recommendation Approach based on User Browsing Behavior

Most traditional recommender systems focus on the objective of improving the accuracy of recommendations in a single domain. However, preferences of users may extend over multiple domains, especially in the Web where users often have browsing preferences that span across different sites, while being unaware of relevant resources on other sites.

This chapter presents a collective two-stage approach for generating cross-domain recommendations, hybridly exploiting the semantic content of Web resources in combination with patterns of user browsing behavior. In the first stage of the approach, we present a technique for learning the relevance of resources and predicting which ones are the most relevant to recommend to a user. The second stage comprises a trade-off scheme between resource relevance and diversity, which helps to increase diversity while keeping relevance uncompromised.

We demonstrate the effectiveness of the proposed approach through various experiments with real-world datasets of semantically-enriched logs of user browsing behavior at multiple Web sites.

5.1 Introduction

Recommender systems typically focus on items in a single domain, for example suggesting related books to a user who is currently viewing information about a book, or a list of movies when she is visiting a Web page about a film. The primary objective has mainly been the improvement of the relevance of recommendations.

Recently, there is a growing awareness [Fernández-Tobías et al. 2011, Cremonesi et al. 2011, Loizou 2009] that the interests and needs of users span across different application areas. An emerging trend is the development of cross-domain recommender systems. In the Web, this task is more challenging than in the traditional single domain recommender setup because there is a need to link items (resources in Web parlance) across Web sites. As such, we can expose visitors to novel resources by recommending pages belonging to diverse domains in terms of the type of information contained in the page.

Previous works have already observed that diversification helps to (1) mitigate the cold-start problem, (2) address the sparsity problem, and also (3) provide richer user experience and higher engagement.

Despite an increasing acknowledgment of the necessity for cross-domain recommendations, this research field is still new. A significant challenge in building such recommender systems is the small overlap between the users and resources of different domains. In the Web context, the challenge is to understand the type of resource(s) presented in a Web page and how they relate to resources presented in other pages. In other words, there is a need for *semantic* approaches that can extract the *type* of resources presented in a Web page and apply knowledge (an ontology) about the relations of resources and resource types.

In this work[1], we address the problem of generating recommendations of Web resources across domains based on a collective approach. In Figure 5.1, we graphically illustrate how the approach presented in this chapter is positioned with respect to the overall thesis framework. This approach is built with data from several domains and makes joint recommendations for such domains. We exploit ontological information extracted from Web pages to find relations between resources and es-

[1] The work presented in this chapter was conducted during a research stay at Yahoo Labs, Barcelona.

tablish in this way bridges among domains. We present a new model for learning the relevance of resources and predicting which are the most relevant ones to recommend next to a user, given that the user is currently at a certain page.

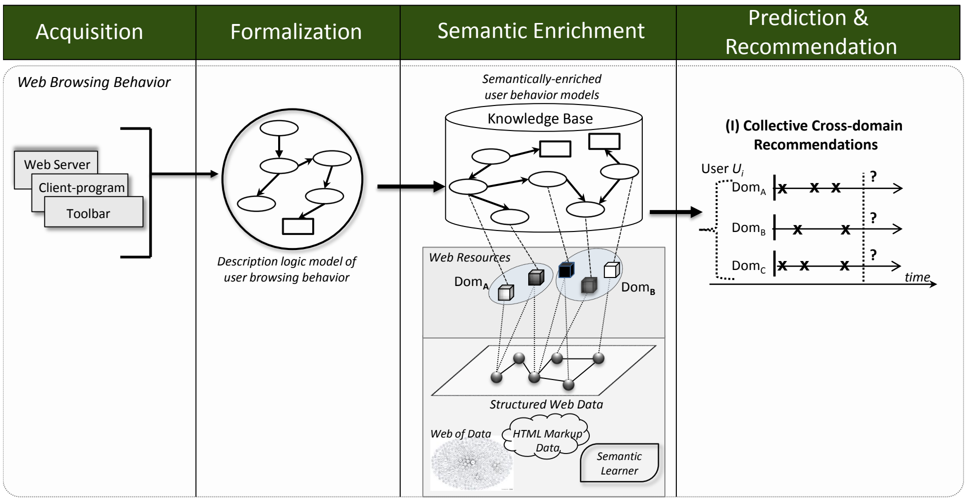


Figure 5.1: Framework overview: collective cross-domain recommendation approach

We further introduce a second stage in the recommendation approach, which ensures that the final recommendations are not only *relevant* to Web visitors, but also highly *diverse* across domains. To reach a balance in the trade-off between relevance and cross-domain diversity, we present a diversification method with a novel formulation of diversity maximization as a knapsack problem.
We evaluate our approach using real world usage data from navigation logs collected with a browser toolbar. We show that our method successfully exploits the semantics of resources and outperformes other popular recommender systems.

5.1.1 Research Questions and Contributions

In this chapter, we address the following research question:

Research Question 2. *Is it possible to provide a predictive method that captures implicit user preferences and the enriched semantics of Web resources in order to generate accurate recommendations of resources across domains?*

In order to properly answer this research question, we need to tackle several challenges that are addressed in the following auxiliary questions:

- User and item profiles are distributed in multiple domains: how to establish a mechanism to bridge these domains?
- How to exploit the structured contextual knowledge about the Web resources?
- How to quantify a measure of relevance of the resources to be recommended?
- How to exploit user preferences implicit in the logs together with the semantic descriptions in order to predict what is relevant to a user?
- How to use the predicted relevant resources in order to generate accurate recommendations?
- How to quantify a measure of cross-domain diversity in order to estimate whether the recommendation list contains resources lying in various domains?
- How to increase the chances that a user is provided with a highly diverse list of cross-domain recommendations? How to effectively maximize this diversity without comprimising the relevance of resources?

The investigation of this research question led to the following main contribution:

Contribution II. *Collective-based technique to generate accurate top-N recommendation of Web resources across domains.*

This contribution is sustained through a set of methods summarized below:

- Semantic recommendation approach, which exploits in novel ways the semantic structures of Web pages and their combination with usage patterns inherent in browsing logs.

  This is a crucial component of our work and it presents a novel contribution in the field of recommender systems. Contrary to existing approaches that generally consider explicit user ratings as usage-based features, our challenging setting contains only implicit preference feedback inherent in user browsing logs. We introduce a probabilistic approach to assess users preference and map them to measurable quantities.

- Model of domain diversity bound to cross-domain recommendations. We present a novel approach for enhancing diversity of resources across domains with a new formulation of the maximization problem based on dynamic programming.

- First real-world study on (1) leveraging user behavior at browsing *multiple* Web sites with structured markup data, and (2) the impact of structure in making joint cross-domain recommendations to Web users.

5.1.2 Outline

We present our problem statement in Section 5.2 and introduce the cross-domain recommendation framework in Section. 5.3. The relevance prediction method is presented in Section 5.4, while the diversity enhancement technique is explained in Section 5.5. We have conducted extensive experiments to evaluate our approach. The experimental setups and evaluation results are shown in Section 5.6. After referring in Section 5.7 to a set of works related to ours, we finally draw conclusions in Section 5.8.

5.2 Problem Statement

Positioned in a cross-site browsing scenario, our task is to recommend to users the top-N pages that they might visit next. We consider a recommendation setting in which user and item profiles are distributed in multiple domains.

We adhere to the definition [WINOTO and TANG 2008] of a domain as the set of similar items with the same characteristics that can be easily differentiated, e.g. movies, concerts, songs, news, artists, etc. Also, since the term “category“ or “type“ is sometimes used, we will use them interchangeably with “domain“.

Our goal is to suggest to a user resources of different types from across the Web, not just from the Web site currently being visited. This scenario is illustrated in Figure 5.2: a user has accessed a page about the *Kings of Leon Concert* in Columbus, at the Web site eventbrite.com. Subsequently, the user has visited the pages about the performer *Kings of Leon* and the performer *Lumineers*.

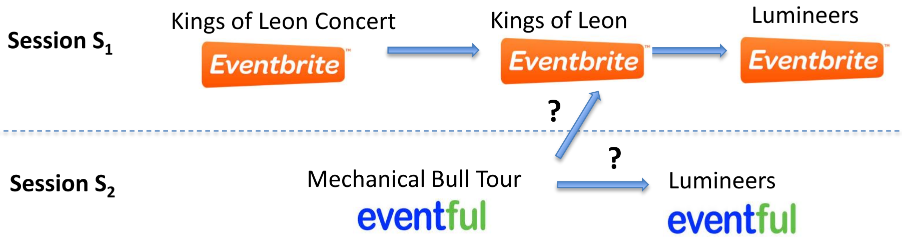


Figure 5.2: Cross-domain recommendation task

In a separate session, another user visits a page of the *Mechanical Bull Tour* at the site eventful.com, which is the same real-world concert as the *Kings of Leon Concert* in the other Web site. Based on the history of the previous user and content of the pages, we want to recommend resources that are relevant, yet from various domains, in order to allow the user to discover new sources of information. For instance, the recommendation for browsing next the page of the performer *Kings of Leon* at eventbrite.com is still relevant, yet of a different type from *concert* and at a different *site* from the one this user is currently visiting. We aim at providing solutions to build a global recommender that suggests related pages to the one the user is currently viewing, and exploits aggregate past behavior of other users.

In the following, we give the definition of our cross-domain recommendation task. Without loss of generality, we define the task when two domains are involved, using the notation of [CREMONESI et al. 2011, FERNÁNDEZ-TOBÍAS et al. 2012].

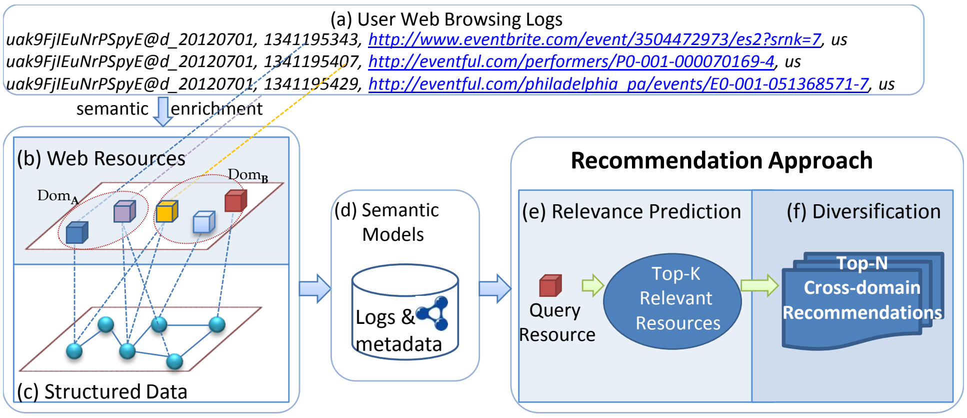
Definition 11. (Collective Cross-domain Recommendation Task) *Let $\mathcal{U}_\mathcal{A}$, $\mathcal{U}_\mathcal{B}$ be the sets of users and $\mathcal{R}_\mathcal{A}$, $\mathcal{R}_\mathcal{B}$ be the sets of resources with characteristics (user preferences and resource attributes) in the domains $\mathcal{A}$ and $\mathcal{B}$, respectively. Our recommendation tasks is to make* joint recommendations *of resources belonging to different domains, i.e., suggesting resources in $\mathcal{R}_\mathcal{A} \cup \mathcal{R}_\mathcal{B}$ to users in $\mathcal{U}_\mathcal{A} \cup \mathcal{U}_\mathcal{B}$.*

There are various types of overlaps between domains that have been identified [CREMONESI et al. 2011]. Our task is conducted in the setting where we might have *user overlap* among domains (i.e. a user browsing various domains), but *no resource overlap* (i.e. each domain has its own resources).

Since user and item profiles are distributed in multiple domains, we have to establish a mechanism to bridge them. To address the non-overlap situation, we make use of the formalization and enrichment approach presented in Chapter 4, which enables us to explore the content of Web pages and find semantic structured description of resources across various domains. We use this foundation to build content-based relations between Web resources that serve as semantic bridges connecting different domains.

5.3 SUADEO: Recommendation Approach

In order to tackle the problem stated above, we introduce a cross-domain recommendation framework consisting of a two-step approach, referred to as SUADEO [2]. We consider sessions of user browsing logs, where a session S consists of a set of events e_n, each representing a visit to a resource $r_i \in R'$ by a given user at a given time. A resource r_i may be linked to a resource in the ontology O. This approach is graphically illustrated in Figure 5.3.



```
uak9FjIEuNrPSpyE@d_20120701, 1341195343, http://www.eventbrite.com/event/3504472973/es2?srnk=7, us
uak9FjIEuNrPSpyE@d_20120701, 1341195407, http://eventful.com/performers/P0-001-000070169-4, us
uak9FjIEuNrPSpyE@d_20120701, 1341195429, http://eventful.com/philadelphia_pa/events/E0-001-051368571-7, us
```

Figure 5.3: Cross-domain top-N recommendation approach based on user browsing behavior

In the first step, we predict a set K of resources as the most relevant ones for a user to visit next, given that the user is at the moment at page r_i (referred as *query*

[2] Suadeo is the latin word for *advise*

resource). In the second step, we apply a diversity enhancement approach on the set K to generate a final set of N recommendations from diverse domains. We have to ensure not to compromise the accuracy of the resources previously predicted as highly relevant.

5.3.1 Discovering Semantic Bridges

A crucial and novel element of our approach is the hybrid mechanism of exploiting semantic information embedded in the content of the Web pages in combination with usage patterns inherent in the user browsing logs. The formalization of usage logs and their semantic enrichment is described in Chapter 4. Our method is open-domain, in that it does not exploit domain-specific heuristics. The process starts with the extraction of user browsing logs (tracked by a client toolbar) and their segmentation into sessions.

The next steps consist in identifying the set of unique pages in the filtered user logs and deploying metadata extraction and metadata analysis techniques. Different Web sites use different schemas to annotate their HTML elements. Hence, we align the concepts and relations among the schemas based on their respective semantics, in order to enable matching resources across different sites. The result is a reference ontology $\mathcal{O}$ with concepts and their semantic relations used for the annotation of resources across all sites.

The resources are classified into different types, i.e. *classes* of the ontology (e.g. *Performer*, *Venue*) and are connected to each-other via semantic *relations* (e.g. *hasPerformer*, *hasVideo*). We further perform entity linking in order to identify resources that belong to different Web sites, but still semantically represent the same object in the real world (e.g. same performer, venue, etc.). This is modeled with the *owl:sameAs* relation. Figure 5.4 illustrates how the example of user logs previously shown in Figure 5.2 are enriched with ontological knowledge. For simplicity, we consider the same example of logs as in Section 4.4.2 of the previous chapter. For instance, the resource *Kings of Leon* of type *Concert* is related to the resource of type *Performer*. The respective pages of the performers in the two different sites represent the same real-world entities.

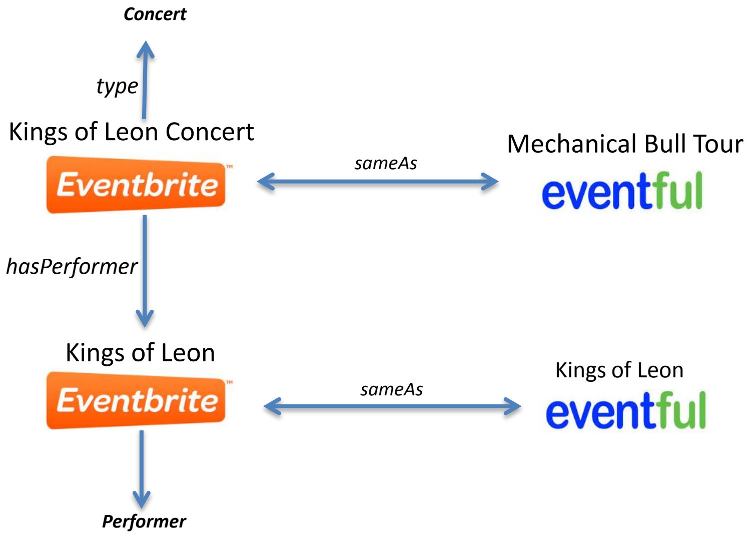


Figure 5.4: Example of the ontological knowledge captured by the semantic enrichment methods

In the following section, we describe how the captured ontological information is used to learn the relevance between resources of various domains.

5.4 Relevance Model

In this section, we define resource *pair relevance* and *set relevance*, then propose an approach to estimate their values. We describe how we use the relevance values to generate the inital recommendation set K.

Pair Relevance. Given two Web resources r_i and r_j, let pair relevance $P(rel|r_i, r_j)$ denote a value that captures the relevance of these resources to each-other. We give a probabilistic interpretation to the relevance values: they approximate the *likelihood* of r_j satisfying the user intent given the query resource r_i. In our case, $P(rel|r_i, r_j)$ is determined by a scoring function based on user access patterns and content of resources r_i and r_j. Pair relevance is an item-item similarity measure, thus the order in the pair is not important.

Set Relevance. We define the relevance of a set of recommendations as:

$$Rel(K|r_i) = 1 - \prod_{j \in K} (1 - P(rel|r_i, r_j)) \tag{5.1}$$

based on an *independence* assumption: given a query resource r_i, the conditional probabilities of two other resources satisfying the user are independent.

The probability that the user will find none of two resources r_j and r_k relevant equals $(1 - P(rel|r_i, r_j))(1 - P(rel|r_i, r_k))$, s.t. $(1 - P(rel|r_i, r_j))$ is the probability that r_j fails to satisfy. The probability that the set K will all fail to satisfy equals its product, by the independence assumption. One minus that product equals the probability that some resource in the set will satisfy the user.

5.4.1 Relevance Learning and Prediction

We formulate the problem of estimating resource pair relevance $P(rel|r_i, r_j)$ as a binary classification task.

Learning Pair Relevance. We apply Support Vector Machines (SVMs) as an established machine learning technique for discriminative classification of high-dimensional sparse data. The task is to learn a decision function $f : \mathcal{R}^d \rightarrow Y$ based on an i.i.d training sample D_{train}=$\{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ..., (\mathbf{x_n}, y_n)\}$, where each training example consists of a feature vector $\mathbf{x} \in \mathcal{R}^d$ and an output label $y \in \{-1, 1\}$. The learned function f is then used to predict the output label $sign(f(\mathbf{x_k}))$ for test example $\mathbf{x_k}$.

Our original input data consist of a set X of resource pairs. Each pair $x = \langle r_i, r_j \rangle \in X$ is initially mapped to a d-dimensional feature vector $\mathbf{x}$ via a function $\psi : X \rightarrow \mathcal{R}^d$. The output labels in $Y = \{-1, 1\}$ denote in our case the two classes:*non-relevant* and *relevant* resources in the pair.

Probability Estimates. For relevance prediction, we are not just interested on hard decisions (labels), but rather the probability $P(rel|r_i, r_j)$ (Eq. 5.1). We formulate it as an estimate of the confidence in the correctness of the predicted label. It is defined as the class conditional posterior probability $P(y|\mathbf{x}) = P(y|\psi(x))$, i.e. the probability with which the feature vector $\mathbf{x}$ of pair $x = \langle r_i, r_j \rangle$ belongs to class y. Therefore, we deploy Support Vector Machines (SVMs) as probabilistic models by further calibrating the scores into an accurate class conditional posterior probability with the sigmoid function [PLATT 2000]:

$$P(rel|r_i, r_j) = P(y = 1|\psi(x = \langle r_i, r_j \rangle)) = \frac{1}{1 + exp^{(Af(x)+B)}} \qquad (5.2)$$

fitted to the decision values of f, with parameters A and B estimated by minimizing the negative log likelihood of training data (using their labels and decision values) [LIN et al. 2007].

Predicting Relevant Resources. The approach allows us to learn a model, which we use to predict a set K of resources that are relevant to a query resource r_i (i.e. have a pair relevance value above 0). We first derive pairs of resource r_i with other resources, then apply the model learned with the SVMs to estimate the relevance $P(rel|r_i, r_j)$ for each pair.

Generating top-N Recommendations. We select the top-N resources from set K with the highest relevance to r_i, ordering by the predicted pair relevance values. This set composes the user recommendations for further Web navigation.
A crucial part of the prediction method is to define for the resource pairs the features that are effective in predicting an accurate relevance value. The novelty of this work is the introduction of features that exploit the semantic information of resources, in order to overcome the problem of lacking overlaps between domains.

5.4.2 Features

We engineer two groups of features: (1) content-based features that use the content of the resources, and (2) usage-based features, which exploit the information contained in the user logs. Some features especially capture the semantics in the structured content of resources.
SEMANTICSIMILARITY: A measure estimated via a set spreading approach [THIAGARAJAN et al. 2008] using the structural information related to the Web resources. Our spreading approach (Fig. 5.5) appends to a resource descrip-

tion terms that are related to the original terms based on an ontology. This is the ontology $\mathcal{O}$ constructed in our semantic enrichment approach (Sec. 5.3.1).

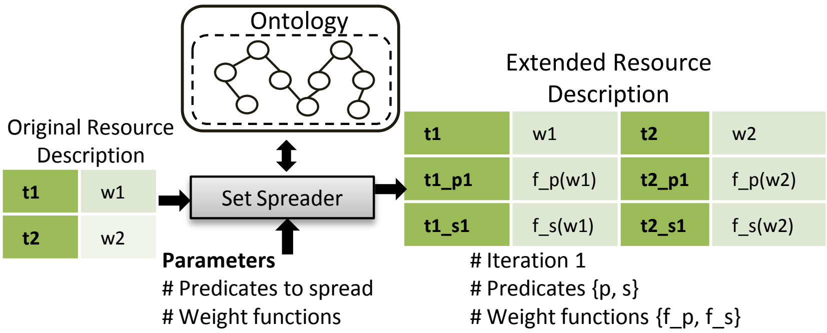


Figure 5.5: Set Spreading Approach

The process starts with an initial set $RD_k = \{\langle t_k, w_k \rangle\}$ for each resource description, where t is the semantic type (e.g. $Performer$, $Event$, etc.) of this resource k denoted by a concept in the ontology $\mathcal{O}$, and weight w_k denotes the importance of the concept term in describing the resource. Each resource description RD_k is then iteratively extended via spreading, utilizing the concepts and relations (predicates) in $\mathcal{O}$. The set spreading of an RD_k results in the resource description $RD'_k = \{\langle t_k, w_k \rangle, \langle t_k\_p, f\_p(w_k) \rangle\}$, which is extended by the term t_k_p related to t_k in $\mathcal{O}$ by the predicate p. We pre-defined a set of predicates to find, at each iteration, $related$ terms of the previous RDs. We use a simple function $f\_p(w_k) = 0.75 w_k$ to estimate weights, reducing them at each iteration.[3] The spreading process is terminated by predicates exhaustion. The final similarity of two resources is then the mean cosine similarity of their descriptions RD_i.

Let's consider a simple example of computing the similarity of the following RDs:

$$RD_1 = \{\langle BrantleyGilbert, 1.0 \rangle\}$$
$$RD_2 = \{\langle ChelyWright, 1.0 \rangle\}$$

[3] The less related the resources are via terms in $\mathcal{O}$, the smaller is their similarity degree at each iteration.

The initial intersection check between the RDs would result in an empty set. Spreading the RDs, by referring to the type relation, extends them to

$$RD'_1 = \{\langle BrantleyGilbert, 1.0\rangle, \langle countrysinger, 0.75\rangle\}$$
$$RD'_2 = \{\langle ChelyWright, 1.0\rangle, \langle countrysinger, 0.75\rangle\}$$

For each RD, we initially start with a default weight of 1. After spreading, the weight of the extended term is decreased at each iteration. The intersection check between the extended descriptions RD'_1 and RD'_2 would result now in a non-empty set, indicating relatedness between the resources. The result of the spreading, which in this example is the inclusion of the related term *country singer*, ensures that any relationship existing between the RDs is taken into consideration for computing their similarity.

SHARETYPE: A binary value indicating if the pair (r_i,r_j) of resources have the same type (concept of the ontology $\mathcal{O}$).

SHARERELATION: A binary value indicating if resources of pair (r_i,r_j) share a relation (in the ontology $\mathcal{O}$) between them, e.g. one resource is the event and the other resource is its venue, therefore sharing the relation *hasVenue*.

The following feature uses no semantics, but is rather used a baseline for item-to-item similarity based only on syntactical representations.

SYNTACTICSIMILARITY: The term vector similarity measure between any two Web resources, computed as the cosine angle between two vectors modeled out of the *bag-of-words* (BOW) representation of the HTML page of each resource:

$$sim_{syntactic}(r_i, r_j) = \frac{\mathbf{V}(r_i) \ \cdot \mathbf{V}(r_j)}{|\mathbf{V}(r_i)| \ |\mathbf{V}(r_j)|} \tag{5.3}$$

$\mathbf{V}(i)$ is a real-valued vector composed of the weights of terms found in the HTML content of resource r_i. The weights are computed using the TF-IDF weighting scheme [MANNING et al. 2010]. As such, this feature entails only syntactic information.

We also define a group of session-based features, which are computed based on the user logs. As described earlier, we consider the implicit preference judgments of Web users captured in their interaction with the system (*clicks*). Since no explicit ratings are available, it is challenging to translate these preferences into measurable usage patterns in order to provide a collaborative-based filtering approach.

OBSERVEDRELEVANCEDEGREE: This measure captures observations from usage patterns in the sessions of browsing logs. We model the correspondence between resource usage counts and user interest as a heuristic mapping between the access patterns and the probability of relevance. We adapt the Expected Reciprocal Rank metric [VARGAS and CASTELLS 2011a] for the setting of aggregated user sessions, introducing the scheme:

$$ORD(r_i, r_j) = \frac{2^{\mathrm{g(i,j)}}}{2^{\mathrm{g\_max}}} \tag{5.4}$$

$$\mathrm{g}(r_i,r_j) = \mathrm{n} \cdot \mathcal{F}(Freq(r_i, r_j)) \tag{5.5}$$

$$\mathcal{F}(Freq(r_i, r_j)) = \frac{|\{r_k \in S' | Freq_S(r_i, r_k) \leq Freq_S(r_i, r_j)\}|}{|S'|} \tag{5.6}$$

where $S' \subseteq S$. The value $g(r_i, r_j)$ denotes the observed URL access frequencies in the overall user sessions. It is normalized to a common rating scale $[0, n]$, based on cumulative distribution function of $Freq(r_i, r_j)$ over the set of other URLs accessed in the same session with r_i and r_j, but co-located with r_i less frequently. The maximum relevance value is g_max.

CONDITIONALSIMILARITY: Conditional probability of pair (r_i, r_j) occurring in the same session, given that resource r_i appears in that session:

$$sim_{conditional}(r_i, r_j) = \frac{Freq_S(r_i, r_j)}{Freq_S(r_i)} \tag{5.7}$$

SESSIONSIMILARITY: A binary value stating if the two resources in the pair (r_i, r_j) appear together in at least one user session from the set S.

$$sim_{session}(r_i, r_j) = \begin{cases} 1, & \text{if } Freq_S(r_i, r_j) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.8}$$

where $Freq_S(r_i, r_j)$ is the number of sessions in which resources r_i and r_j occur together.

5.5 Diversity Model

Despite the various perspectives to approach diversification, recent works [ZIEGLER et al. 2005, AGRAWAL et al. 2009, FERNÁNDEZ-TOBÍAS et al. 2011] seem to agree on linking diversity to the topics/categories of a particular taxonomy. We also use the notion of resource category in our definition of diversity, but also exploit the aspect of Web site diversity. The resource category is represented in our case by the *class type* which semantically annotates a Web resource. Our definition of diversity (Def. 12) among resources covers two aspects: (1) semantic *type* of resources, and (2) *Web site* where they are located.

Definition 12. (Diversity of Resources) *Two Web resources in pair (r_i, r_j) are diverse if any of the following conditions occurs: (1) they have different semantic* types, *(2) they are located in different Web sites.*

For a measurable estimation, we define a distance function that considers both aspects of diversity in Def. 12. Specifically, given a set K of *relevant* resources for query resource r_i, to produce the final list of recommendations R, we measure the distance between any two recommended resources $r_j, r_l \in K$ by the function $d : K \times K \to R$, s.t.

$$\begin{aligned} d(r_j, r_l; w_t) = \ & w_t f_{\texttt{type}}(r_j, r_l) + (1 - w_t) f_{\texttt{site}}(r_j, r_l) \\ & \qquad s.t.\ \ f_{\texttt{type}}(r_j, r_l) \in \{0, 1\}, f_{\texttt{site}}(r_j, r_l) \in \{0, 1\} \end{aligned} \tag{5.9}$$

The symmetric distance function is the weighted average of the binary values produced by functions f_{type} and f_{site} that, respectively, define if the *type* or the *site* of resources match. The weight w_t denotes the importance of diversity in *type*. It can be configured according to the application at hand.

The overall diversity of a set of resources is modeled as the *average dissimilarity* of all pairs of resources in the set. We use the averaged intra-list distance metric [VARGAS and CASTELLS 2011a]:

$$Div(R) = \frac{1}{|R|(|R|-1)} \sum_{r_j \in R} \sum_{r_l \in R, r_j \neq r_l} d(r_j, r_l) \tag{5.10}$$

where, in our case $d(r_j, r_l)$ is the distance function in Eq. 5.9.

5.5.1 Diversity Enhancement

The initial set of recommendations is constructed with a relevance maximization method that follows a similarity-based approach (Sec. 5.4). As such, the rationale behind enhancing the diversity of the recommendation set is that the resources selected as recommendations are very likely to be similar to each other. If the final set comprises diverse Web resources, it is more likely that a user finds in this set those recommendations that fulfill her navigation intent.

However, an approach that suggests to the user diverse, but non-relevant resources is not able to offer satisfactory results. Therefore, the goal of jointly offering a final set R of recommendations with high diversity and high (similarity-based) relevance are opposite to each-other. To enhance diversity, we follow the Maximal Marginal Relevance (MMR) scheme [CARBONELL and GOLDSTEIN 1998] by maximizing a trade-off objective between the relevance and diversity of the recommendation set:

Given a query resource r_i, a set K of resources predicted as relevant to r_i, integer N, and fixed control parameter $\lambda \in [0, 1]$, find the set of resources $R \subseteq K$ with $|R| = N$ that maximize the objective function:

$$\mathcal{F}(R, N, \lambda) \triangleq (1-\lambda)\boldsymbol{Rel(R|r_i)} + \lambda \boldsymbol{Div(R)} \tag{5.11}$$

The objective in Equation (5.11) is the weighted, linearly normalized, arithmetic mean of set relevance and set diversity. The parameter λ controls the degree of trade-off.

Maximization Algorithm

A *greedy* method is often used to search the optimal subset that satisfies a given objective, making a locally optimal choice at each stage, hoping to produce a solution that approximates the global optimum. On the other side, a brute-force combinatorial approach is effective in finding the global optimal solution, but computationally infeasible. We present an approach with a novel formulation of the diversity optimization problem, which is efficiently solved by dynamic programming (DP).

We formulate the maximization task as a variation of the *0-1 Knapsack Problem*, where resources from a set K have to be packed in a knapsack of capacity N. Each resource has a fixed weight w_j and a quality value q_j resulting from the trade-off function f_t (Eq. 5.11). The objective is to pack the knapsack such that we achieve the maximum total quality value of packed resources. Given parameters λ ∈[0, 1] and $N \in Z^+$, we formulate the following maximization problem:

$$\begin{aligned} \text{maximize} \sum_{j=1}^{|K|} \mathcal{F}(K, N, \lambda) x_j \\ \text{subject to} \sum_{j=1}^{|K|} w_j x_j = N \\ x_j \in \{0, 1\}, r_j \in K \end{aligned} \tag{5.12}$$

Our constraints are that each item is chosen *at most once* and the packed items have to fill the knapsack at its *exact* capacity N. In order to solve this maximization problem efficiently, we propose an algorithm based on dynamic programming.

In dynamic programming, the final solution can be recursively described in terms of solutions to subproblems (optimal substructure). The requirements are to have (1) an optimal substructure, s.t. an optimal solution to the problem consists of optimal solutions to subproblems, and (2) overlapping subproblems.

Algorithm 5: Diversity Maximization Algorithm

```
Input: Set $K$, parameter $\lambda$, $N < |K|$, query resource $r_i$
Output: Subset $R \subseteq K$ that maximizes $f_t$ s.t. $|R| = N$
    Initialize set R = $\emptyset$; Indeces matrix $I[|K|, N]$ = $\emptyset$;
    for $w$ = 0 to $N$ do
        $B[0, w]$ = -$\infty$
    end for
    for $j$ = 1 to $|K|$ do
        $B[j, 0]$ = 0
    end for
    for j=1 to $|K|$ do
        for for w=0 to $N$ do
            $w_j$=1
            if $(w_j <= w)$ then
                $R*$ = 0
                for index $d \in I[j-1][w-w_j]$ do
                    $R* \leftarrow (R* \cup r_d)$ s.t. $r_d \in K$
                end for
                $p_j = 1 - (1 - Rel(R*—i))*(1 - P(rel|i, r_j))$
                $q_j = (1 - \lambda)p_j + \lambda Div(R* \cup r_j)$ s.t. $r_j \in K$
                if $(q_j > B[j-1][w])$ then
                    Increase the value which is to be maximized
                    $B[j][w]$ = $q_j$;
                    $I[j][w] \leftarrow I[j-1][w-w_j] \cup j$
                else
                    $B[j][w]$ = $B[j-1][w]$
                    $I[j][w]$ = $I[j-1][w]$
                end if
            else
                $B[j][w]$ = $B[j-1][w]$
                $I[j][w]$ = $I[j-1][w]$
            end if
        end for
    end for
    for $d \in I[|K|][N]$ do
        $R \leftarrow (R \cup r_d)$ s.t. $r_d \in K$
    end for
```

We have to fill the knapsack with N items, which are chosen from a set K of items. Our definition of the subproblem is to compute $B[j, w]$, which is the maximal quality value of items that can be placed in the knapsack. The parameter w denotes the exact weight for each subset R_j of items, whereas the variable j iterates through the values 0 to $|K|$. The quality value is computed with the trade-off function $\mathcal{F}$ (Eq. 5.11). We use the following recursive formula for the subproblems:

$$B[j,w] = \begin{cases} B[j-1,w], & \text{if } w_k > w \\ \max(B[j-1,w], B[j-1,w-w_j] + q_j), & \text{otherwise} \end{cases} \tag{5.13}$$

It means that the best subset R_j, which has total weight w is:
1) the best subset of R_{j-1} that has total weight w, or
2) the best subset of R_{j-1} that has total weight $w - w_j$ plus the item r_j with quality value q_j.
These two main cases are also presented in Algorithm 5. First case, item r_j cannot be part of the solution, since if it was, the total weight would be greater than N (Alg. 5, line 26-29). Second case, item r_j can be part of the solution, and we choose the item with the greater quality value (Alg. 5, line 15-21). As mentioned earlier, the quality value is computed with the trade-off function of relevance and diversity (Alg. 5, line 16-17).
To facilitate the calculation of the quality values, we store in matrix I the indices of the items selected in the previous subproblem (subset R*) (Alg. 5, line 12-15). Note the initialization of first row in B with value $-\infty$ (line 2-4), to solve our constraint of filling the knapsack at its exact capacity.

Complexity. While the Knapsack problems belong to the family of NP-hard problems, the algorithm using dynamic programming provides a *pseudo-polynomial* solution $O(K \cdot N)$. It is exponential in the *length of input* N, hence $O(K \cdot 2^{bitsN})$.

5.6 Experimental Results

5.6.1 Experimental Setup

Dataset of Semantically-enriched Logs. For our experiments, we use datasets of real-world logs that combine cross-site user browsing logs with semantic metadata embedded in the Web pages. This is the dataset presented in Section 4.6.2, which resulted from the formalization and semantic enrichment of Yahoo! toolbar logs recorded in a five-week period. As explained earlier, we selected user logs from three different Web sites: Eventful[4], Eventbrite[5], and Upcoming[6].

We summarize this dataset in Table 5.1. It contains the browsing events of 494 users segmented into 526 sessions, only 2.58% of which are cross-site sessions, i.e. contain events distributed across two or more sites.

	Logs dataset
Logs period	01.Jul.2012 - 07.Aug.2012
Sites	eventful.com upcoming.yahoo.com eventbrite.com
Nr. users	494
Nr. browsing events	2244
Nr. unique URLs	1683
Nr. user sessions	526
Fraction cross-site sessions	2.58%
Metadata formats	RDFa Microdata Microformats

Table 5.1: Statistics of the semantically-enriched toolbar logs dataset

[4] http://eventful.com

[5] http://eventbrite.com

[6] http://upcoming.yahoo.com

An initial analysis of browsing logs shows that users do not stay long in a particular site, leading to small session lengths. We notice that the majority of the users are referred to the sites by search engines, social network sites, or email pages, afterwards staying within our sites of interest to visit very few (mostly, 1 or 2) pages. As such, we selected for our dataset the users that had visited at least three pages from our sites of interest.

We performed another analysis of the browsing logs regarding the frequency with which the URLs are accessed by the users in our dataset. Figure 5.6 plots this frequency distribution, clearly demonstrating the long-tail of URLs visited very rarely, i.e. 76% of URLs are visited just once, and only 1% of them are accessed 5 or more times. The finding motivates the need to increase recommendations diversity, in order to expose the users to the URLs in the long tail.

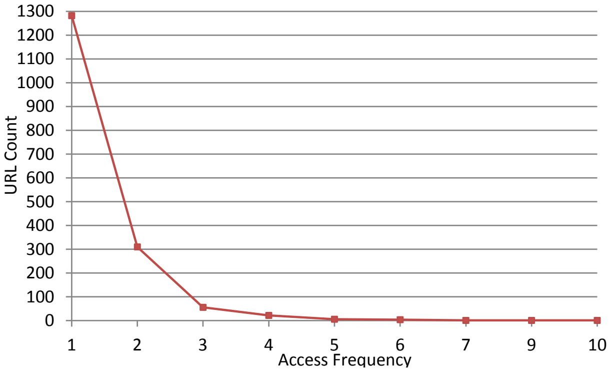


Figure 5.6: Long tail URL access distribution

Ground-truth Dataset. While clicks in the logs are signals of users preferences, we further acquired from human judges ground-truth values for evaluating resource relevance. Initially, we filtered the unique set of resources in the user logs, then extracted a large subset of these resources to pair up among each other. We did not perform a random extraction, rather selected resources by preserving a uniform distribution as in the original set, i.e. keeping the same proportion of resources from the different sites as in the logs. Table 5.2 shows the statistics of this *labeled dataset*.

We showed the pairs (via a Web interface) to human judges, asking them for relevance feedback. They had to decide if the resources in the pair are relevant to

	Ground-truth dataset
Nr. Resource Pairs	1230
Nr. unique Resources	387
Nr. Judges	13
Nr. Judgments/Pair	3
Inter-rater Agreement	80.2%
Non-Relevant Pairs	943 (76.67%)
Relevant Pairs	287 (23.33%)

Table 5.2: Statistics of the ground-truth dataset

each-other (i.e. if after visiting one resource, they would find the other relevant to view next).

Methodology and Evaluation Metrics

We performed three types of experiments for a thorough evaluation of our proposed approach, referred to as SUADEO :

- **Experiment Group 1.** We evaluate the relevance prediction approach using the ground truth dataset (Table 5.2). This dataset is used for training a relevance model with SVMs and testing the results using 10-fold cross-validation.

  Metrics. The decision made by the binary classifier, which labels examples as positive (P) or negative (N), can be represented in a structure that is known as confusion matrix or contingency table (see Table 5.3). In our case, positive and negative correspond to the classes relevant or non-relevant, respectively.

  The confusion matrix is composed of four categories: True positives (TP) refer to examples correctly classified as positives. False positives (FP) are negative examples incorrectly labeled as positive. True negatives (TN) correspond to negative examples, which are in fact correctly labeled as negative. False negatives (FN) are positive examples incorrectly labeled as negative.

	actual positive	actual negative
predicted positive	TP	FP
predicted negative	FN	TN

Table 5.3: Confusion Matrix

Given the confusion matrix, we can define the metrics for evaluating the classifier. As such, *precision* measures the fraction of examples classified as positive that are actually truly positive (Equation 5.14). *Recall* measures the fraction of positive examples that are classified correctly (Equation 5.15). F1 score is the harmonic mean of precision and recall (Equation 5.16).

The True Positive Rate (TPR) is the same as recall, whereas the False Positive Rate (FPR) measures the fraction of negative examples that are misclassified as positive. The Receiver Operating Characteristic (ROC) curve plots TPR (y-axis) as a function of FPR (x-axis). It measures the performance of a binary classifier system at various values of its discrimination threshold. The Area Under the ROC Curve, also referred to as AUC, is the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one.

$$\text{Precision} = \frac{\text{TP}}{\text{TP + FP}} \tag{5.14}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \tag{5.15}$$

$$\text{F1} = \frac{\text{2TP}}{\text{2TP + FP + FN}} \tag{5.16}$$

$$\text{True Positive Rate} = \frac{\text{TP}}{\text{TP + FN}} \tag{5.17}$$

$$\text{False Positive Rate} = \frac{\text{FP}}{\text{FP + TN}} \tag{5.18}$$

Mean Absolute Error (MAE) measures how close the overall classifications are to the actual labels of the examples (Equation 5.19).

$$\text{Mean Absolute Error} = \frac{1}{K}\sum_{i=1}^{K} |f_i - y_i| \tag{5.19}$$

MAE is calculated as the average of the absolute errors $e_i = |f_i - y_i|$, where f_i is the prediction and y_i the actual value, and K is the total number of examples.

- **Experiment Group 2.** To assess our diversification method, we performed experiments on the recommendations generated before and after diversity enhancement. We check how the diversification of results affects the quality of recommendations in terms of the set relevance metric.

  Our Knapsack maximization approach is compared to *greedy* in terms of ranking the top-N recommended resources using two standard evaluation measures for graded relevance values: Normalized Discounted Cumulative Gain (nDCG@N) [BURGES et al. 2005] and Expected Reciprocal Rank (ERR@N) [CHAPELLE et al. 2009].

  Metrics. Most evaluations of Web retrieval systems use cumulative gain-based metrics that support graded relevance. They quantify the usefulness, referred to as *gain*, of a document based on its position in the result list. To compute such metrics, we need the list of results generated by the system and editorial relevance judgments regarded as ground-truths.

  An established measure of ranking quality is the Discounted Cumulative Gain (DCG) [JÄRVELIN and KEKÄLÄINEN 2002], which is very often used when judging the relevance of Web documents. This metric is based on the notion that systems ranking highly relevant documents high in the list are better than systems that rank highly relevant documents low in the list. As such, the gain is accumulated from the top of the list to the bottom, with the gain of each result being discounted at lower ranks.

When we need to evaluate only the top-N results in the list, a useful measure is the DCG at rank N [BURGES et al. 2005] for a given query, which is computed as:

$$\text{DCG@N} = \sum_{i=1}^{N} \frac{2^{rel_i} - 1}{log(i + i)} \tag{5.20}$$

where rel_i is the relevance grade of the document at rank i. While the numerator rewards documents with large relevance grades, the denominator discounts the gains at lower ranks.

Normalized Discounted Cumulative Gain at N (nDCG@N) is a metric based on DCG, which normalizes across queries the cumulative gain at each position for a chosen value of rank i.

For a query, nDCG@N is computed as $\frac{DCG@N}{IDCG@N}$, where the Ideal DCG (IDCG) is the maximum possible DCG until position N. The nDCG values for all queries are then averaged in order to obtain a final measure of the system's performance. In a perfect ranking algorithm, the DCG@N will be the same as the IDCG@N producing an nDCG of 1.0. Therefore, nDCG calculations are relative values on the interval $[0.0, 1.0]$.

In our experiments, we consider as IDCG the top-N relevance values computed with the SVM-based prediction approach, which is itself trained on ground-truth values of the editorial judgments.

Expected Reciprocal Rank (ERR) is a metric for graded relevance, which besides the rank position considers the relevance of results above a document of interest. ERR is defined as the expected reciprocal length of time needed by a user to find a relevant document [CHAPELLE et al. 2009]. It implicitly discounts results shown in ranks lower than very relevant results.

Let $R_{d(i)}$ denote the probability with which the i-th document satisfies the user. It measures the probability of user clicking it, which can also be interpreted as the relevance of the document. In our experiments, it is bound to the relevance measure of our model (Section 5.4). With N denoting the total number of documents in the ranking, then ERR is computed as:

$$\text{ERR@N} = \sum_{r=1}^{N} \frac{1}{r} \prod_{i=1}^{r-1} (1 - R_{d(i)}) R_{d(r)} \tag{5.21}$$

Furthermore, we test in another set of experiments the *scalability* of our maximization method. We report on the average execution time of 140 different queries, after having repeated the experiments three times. Recent works on diversity focus more on recommendation quality than speed, but since diversification can be an expensive task, it is useful to investigate it further.

- **Experiment Group 3.** We compare our recommendation approach to other baselines, applying the metrics of 1-call@N and precision P@N [SHI et al. 2012a] on the ground-truth dataset. 1-call@N is the ratio of test queries for which we find at least one relevant item in their respective top-N recommendation list. The measure P@N reflects the ratio of the number of relevant resources in the top-N recommended resources.

Parameter Setting. For all the experiments, the weight parameter of the distance function (Eq. 5.9) is $w_t = 0.5$; control parameter $\lambda = 0.5$, unless reported otherwise (Sec. 5.6.3).

5.6.2 Results of Relevance Prediction

Experiment I. Relevance Prediction

We evaluate the performance of the SVM-based prediction approach applying different groups of features: usage-based, syntactic content-based, and semantic content-based only, as well as content-based and usage-based features combined. The results are illustrated in Table 5.4. A graphical representation of these results is illustrated in Figure 5.7.

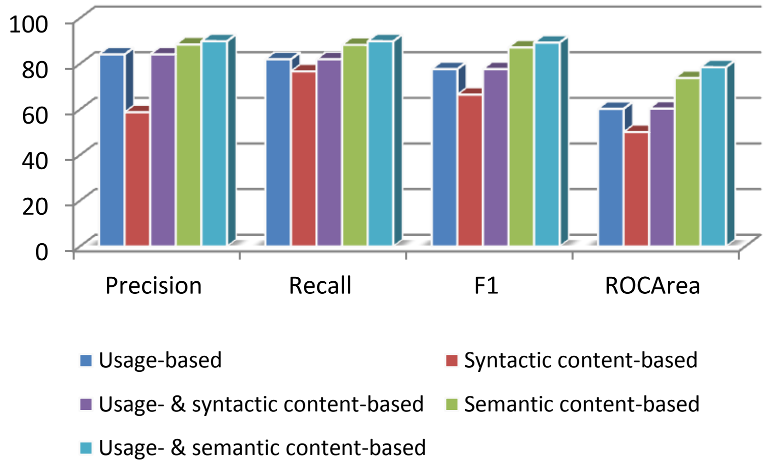


Figure 5.7: Results of relevance prediction: hybrid approach outperforms the others across all measures

We achieve a large improvement when using semantic features in addition to syntactic content-based features. This illustrates the advantage of capturing the structural information of the content when predicting relevant resources. The approach that combines both usage-based and semantic content-based features improves the results, outperforming the others across all measures (i.e. higher precision, recall, ROC Area, and smaller errors).

Features	Precision	Recall	F1	ROCArea	Mean Abs. Error	Root Rel. Sqr. Error(%)
Usage-based	84.1	82	77.7	60.3	0.2935	90.5532
Syntactic content-based	58.8	76.7	66.5	50.1	0.3578	99.9
Usage- & syntactic content-based	84.1	82	77.7	60.4	0.2935	90.5591
Semantic content-based	88.5	88.3	87.2	73.8	0.2075	76.0892
Usage- & semantic content-based	**89.9**△	**89.9**△	**89.3**△	**78.5**△	**0.1812**▿	**71.0287**▿

Table 5.4: Results of the relevance prediction approach using different sets of features. Bold numbers indicate an improvement (difference ≥ 0.1). The sign △ marks the highest value, whereas ▿ the lowest value.

5.6.3 Results of Diversity Enhancement

Experiment II. Relevance and Diversity Trade-off

In this experiment, the goal is to check whether the diversity maximization approach sacrifices relevance to achieve diversity. We observe the measures of *set relevance* (Eq. 5.1) and *set diversity* (Eq. 5.10) before and after diversification. We respectively refer to the approaches as SUADEO and SUADEO$^{\text{DIV}}$, that is the approach with diversity enhancement.

We perform tests with different values of the control parameter λ. The results presented in Figure 5.8 illustrate the trade-off between relevance and diversity, by plotting λ vs. set diversity, and λ vs. set relevance with fixed $N = 3$.

The following observations can be drawn from the results: first, the overall diversity of the recommendation set generated by SUADEO$^{\text{DIV}}$ increases significantly when compared to the baseline without diversification. Most importantly, this is achieved without deviating from the set relevance measured before diversification. The relevance of the final recommendations of SUADEO$^{\text{DIV}}$ is very close to the value before diversification for λ up to 0.6. When λ increases further, we see that the relevance of the set starts to decrease.

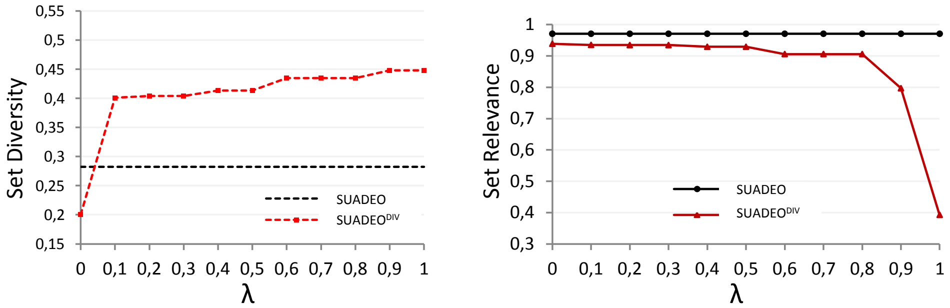


Figure 5.8: Relevance and diversity trade-off in terms of mean set diversity and mean set relevance

The observations indicate that we can improve recommendations quality by controlling λ, as we see that for large values, set relevance decreases significantly while there is no more gain on diversity.

Experiment III. Effectiveness of Diversification

Another experiment investigates the effectiveness of our proposed *knapsack* maximization approach, comparing it to the traditional *greedy* heuristic. The goal is to observe how both approaches differentiate with respect to the ranking of the resources in top-N positions. Note that as graded relevance scale we use the scores obtained from SVMs, trained on the human-labeled dataset. Figure 5.9 shows the results in terms of nDCG@N and ERR@N for different values of N.

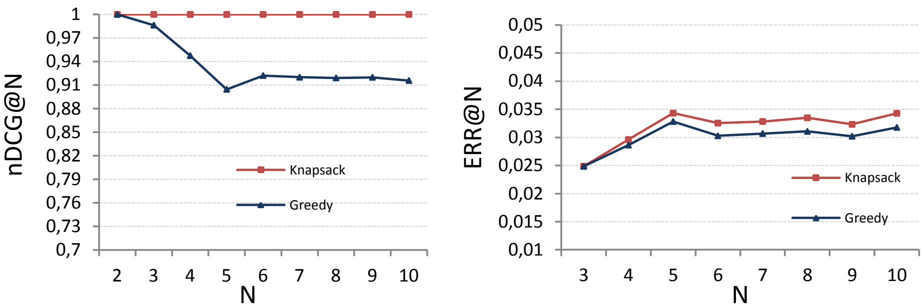


Figure 5.9: Effectiveness of knapsack maximization

The first observation is that the proposed *knapsack* maximization outperforms *greedy* across both measures. *Greedy* positions relevant resources in lower ranks of the recommendation list. The results achieved by *knapsack* are particularly better for increasing values of N($\geq$ 5). Overall, *knapsack* achieves high nDCG@N, being able to rank relevant resources higher in the list. The results indicate a behavioral difference among the two approaches with respect to their ability to effectively position the resources in top-ranked recommendations.

Experiment IV. Scalability

We compare the scalability of the proposed *knapsack* maximization algorithm to the *greedy* approach (Fig. 5.10). The goal is to check whether *knapsack* ensures accuracy at high computational costs. The results indicate that both *greedy* and *knapsack* scale similarly. For N up to 5, *knapsack* takes the same time as *greedy*

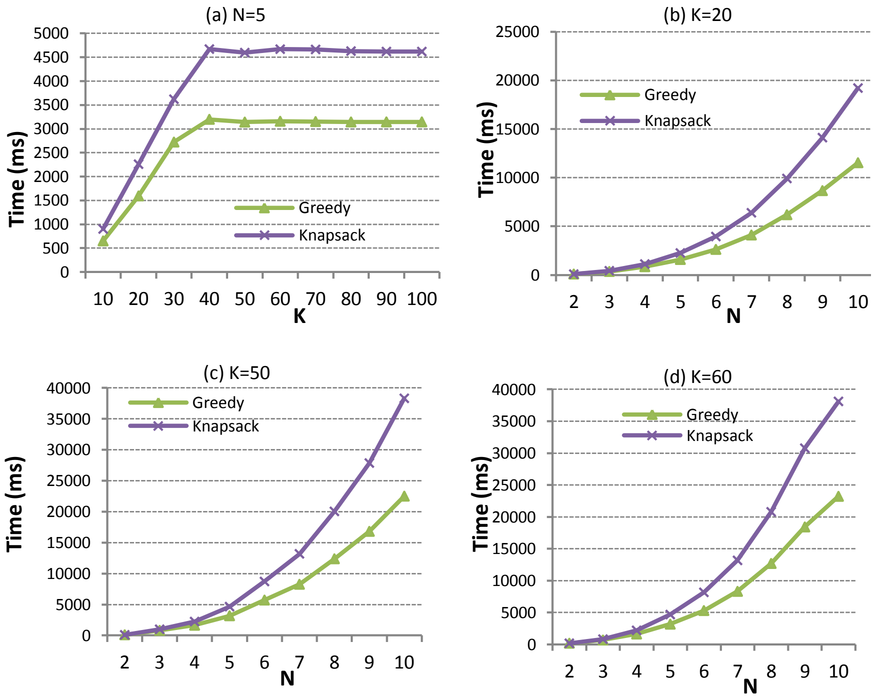


Figure 5.10: Scalability of the knapsack maximization algorithm compared to the greedy heuristic

for diversification. For both approaches, time gradually increases for larger sets K. Yet, at $|K| = 40$ convergence is reached and time does not change for larger sets upon which we diversify.

5.6.4 Results of Recommendation Performance

Experiment V. Comparison of Recommendation Performance

We compare the performance of our recommendation approach with other baselines:

- **UBCF**: User-based collaborative filtering recommendation approach [MOBASHER et al. 2003] combined with content-related features that applies matrix factorization. We use this approach as representative of traditional recommendation approaches that combine collaborative filtering and content-based features. It finds k similar items (neighbors) that are co-rated (or visited) by different users similarly. For a target item, predictions can be generated by taking an average of the target user's item ratings (or weights) on these neighbor items. Since we also deal with usage data, instead of rating we use an implicit binary weight associated to an item (i.e. Web resource) in a user session.

This weight is binary, representing the existence or non-existence of the item in the user session. In our experiments, we set k to 20. We extract resource pairs out of per-user recommendations: for each query resource, we find those users in the logs that have visited it, then generate the top-ranked recommendations for each user. The final list consists of top-N recommendations ranked across all the filtered users.

- **IBCF**: Classical item-based collaborative filtering recommendation algorithm[7]. This algorithm analyzes the user-item matrix to identify similarity relations between the different co-rated/ co-visited items, then uses these relations to compute a list of top-N recommendations. In our experiments, an item corresponds to a Web resource, whereas item-item similarites are computed with the cosine function of the TF-IDF vectors constructed from the HTML content of each resource.

  As in the UBCF approach, we initially extract resource pairs out of per-user recommendations. For each query resource, we identify the users who have accessed it in the logs. Afterwards, we refer to the user-item matrix and apply the item-based CF algorithm in order to generate the top-ranked recommended resources for each of the identified users. The final list consists of top-N recommendations ranked across all the filtered users.

- **SUADEO**: the proposed semantic-based recommendation approach.

- **SUADEO$^{\textbf{DIV}}$**: Suadeo approach with diversity enhancement.

[7] Mahout implementation: http://mahout.apache.org/

The recommendation performances of SUADEO and SUADEO$^{\text{DIV}}$ and the baseline approaches in terms of 1-call@N and P@N are shown in Figure 5.11. The following observations can be drawn: before diversification, i.e. SUADEO outperforms all other methods in terms of P@N. Since the diversification step introduces new resources in the list, it is expected to lower precision, but is important to note that this degradation is small (≤ 0.02 for P@10). Furthermore, there is an overall improvement over the baselines in terms of P@N. These results corroborate that SUADEO$^{\text{DIV}}$ achieves the goal of keeping relevant resources in the top-N recommendations even after diversity enhancement.

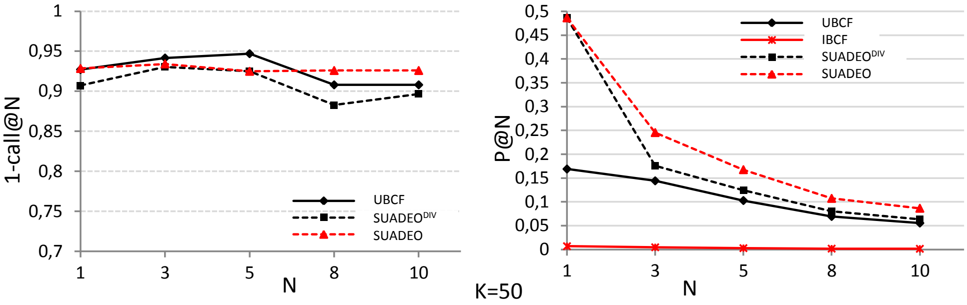


Figure 5.11: Comparison of recommendations quality

In terms of 1-call@N, UBCF gives higher values for smaller N, and is outperformed by SUADEO for $N > 5$. These methods ensure to make at least one recommendation, among a few top-ranked resources, that is indeed relevant to the user. Whereas, IBCF recommender performs poorly with 1-call@N ≤ 0.125 for all N (for clarity not shown in the graph).

In addition, the quality of recommendations generated by SUADEO$^{\text{DIV}}$ does not deteriorate after diversification, observing a small difference $(0.02 - 0.04)$ between SUADEO$^{\text{DIV}}$ and SUADEO , even reaching the same value for 1-call@5. Besides diversity enhancement, SUADEO$^{\text{DIV}}$ contributes by providing relevant recommendations at top-N positions.

In Table 5.5, we show examples of top-4 recommendations generated for a query resource. Given that a user is currently viewing the resource q, which is the venue

	Resource
q	http://eventful.com/oakland/venues/oracle-arena-/V0-001-000515210-3 *Type*: Venue *Description*: Oracle Arena San Francisco, Oakland, CA *PLD*: Eventful
r_1	http://oracle-arena-group-sales-eventful.eventbrite.com/r/eventful *Type*: Event *Description*: Concert in the venue Oracle Arena *PLD*: Eventbrite
r_2	http://eventful.com/oakland/venues/art-soul-/V0-001-004406750-3 *Type*: Venue *Description*: Art & Soul venue in San Francisco *PLD*: Eventful
r_3	http://eventful.com/performers/P0-001-000249063-1 *Type*:Performer *Description*: Maze featuring Frankie Beverly performing in the venue Oracle Arena *PLD*: Eventful
r_4	http://eventful.com/berkeley/events/sahaja-meditation-berkeley-/E0-001-038324733-0@2012070219 *Type*: Event *Description*: Event located in CA *PLD*: Eventful

Table 5.5: Examples of recommendations generated for a query resource

Oracle Arena located in San Francisco, CA, the approach generates a set of recommendations with four resources.

The recommended resources are related to the query resource, for example r_1 is a concert that is organized in that particular venue. Yet, the suggested resources have to be diverse among each-other, e.g. r_1 is of type Event and belongs to the site (PLD) Eventbrite, rather than Eventful. The third resource is a performer, who is performing at that particular venue. The last recommendation is a resource of type Event located in CA, which is also the location of the venue Oracle Arena.

5.7 Related Work

The systems that adopt semantic technologies into recommender systems referred to as semantic recommender systems. Their performance relies on a knowledge base usually defined as a taxonomy or an ontology. We provide a detailed review of these works in Section 3.1, in which we survey state of the art approaches and position this thesis with respect to these contributions.

The majority of semantic recommender systems provide recommendations in a single domain. Cremonesi et al. [CREMONESI et al. 2011] offers a survey of these works. Only few works [FERNÁNDEZ-TOBÍAS et al. 2011, LOIZOU 2009] have recently investigated the task of cross-domain recommendations, while also considering semantic features.

Ongoing work of Fernández-Tobías et al. [FERNÁNDEZ-TOBÍAS et al. 2011] recently introduced a generic framework that, using DBpedia as basis, integrates knowledge from several domains to provide cross-domain recommendations. The framework shows the added-value of using the semantic information of items to link concepts from two domains. This information, extracted from DBPedia as a representative of Link Data repositories, is used to build a weighted directed acyclic graph, which upon a weight spreading mechanism allows the identification of matching items between a target domain and source domain. Yet, this work does not exploit the impact of usage-based features or the dynamics of past user behavior in determining items relevance. Another drawback is that an expert has to identify manually the semantic entities and relations of DBpedia, which can then be used to describe and link the domains of interest.

The majority of cross-domain recommendation approaches deal with collaborative filtering, missing the content-based features [FERNÁNDEZ-TOBÍAS et al. 2012]. Moreover, there are no works that combine these two aspects, while also exploiting the structural representation of the content. A related approach is proposed by Loizou [LOIZOU 2009], which also uses a graph structure to represent relations between domains. Wikipedia is used as a universal vocabulary to provide the semantic information of items from various domains.

In terms of semantics, the approach limits the mapping of an item to a page in Wikipedia, if it exists, otherwise free-form tagging is considered. Hence, the strategy fails to capture the full-fledged structure behind the items' content.

In our approach, we specifically cover this gap and exploit the impact of *semantic structures* in generating recommendations *across domains* in an open setting, without relying on a central knowledge base. We consider both usage-based and content-based features from structured data, showing how their combination improves the recommender results.

Diversity in Recommender Systems. Another line of works [ZIEGLER et al. 2005, ZHANG and HURLEY 2008, HURLEY and ZHANG 2011, VARGAS and CASTELLS 2011b, SHI et al. 2012b, BELÉM et al. 2013] is related to the diversity enhancement part of our approach. These works aim at generating recommendations in a single domain, yet they are relevant with respect to the proposed diversification mechanisms.

A highly desirable aspect of recommendations is to expose users to relevant information that would not have been discovered otherwise. Besides accuracy, a fundamental aspect of recommendation utility is the diversity of the set of items being recommended, which is usually related to how different these items are in comparison to each-other. Related contributions in this area have focused on consolidating the notion of diversity and devising algorithms to maximize it, as well as evaluation metrics to estimate the degree of diversity in the recommendations list.

In one of the earlier works, Ziegler *et al.* [ZIEGLER et al. 2005] present a diversification method, which serves to increase diversity of item topics in the personalized recommendations list. The motivation is to provide a high coverage of items from various topics and reflect a wide spectrum of the user's interests. They propose a similarity metric that applies a taxonomy-based classification. The metric is used to compute intra-list similarity as an overall diversity measure of the recommendation list.

An important contribution of this work is an analysis of the correlation between user satisfaction and metrics of accuracy and diversity, based on large-scale online and offline evaluation experiments. Their results show that the diversified list effects users positively on discovering interest coverage. Yet, the users perceive the degree of diversification applied to a list only to a certain extent, and beyond that they do not notice anymore that the results are diversified. They show that recommendation lists with higher diversification perform worse on accuracy-based measures when compared to non-diversified lists, but the overall users' liking of the list is still stronger.

The work of Hurley and Zhang [HURLEY and ZHANG 2011] on diversity of top-N recommendations served as a good basis for our work. It presents the formulation of intra-list diversity and formal statement of the diversification problem as a joint optimization of two objectives, one reflecting preference similarity and the other diversity of items. These are interpreted as two opposing objectives, thus a trade-off between them is to be established through a set of optimization algorithms. An interesting finding demonstrated by their experiments is the importance of the control parameter to obtain the preferred recommendation performance. In comparison to this work, our approach extends the notion of diversity to cover the cross-domain aspect, and offers a new algorithm for the trade-off optimization.

Motivated by the increasing interest of the research community in addressing diversity and novelty as key utility features of recommender systems, Vargas and Castells [VARGAS and CASTELLS 2011b] point to the lack of standardized methodological and conceptual ground in this emerging field. It is rightly claimed that there exists different evaluation metrics whose relation or distinction need to be defined. There are also various principles used by different works, which would benefit from a methodological unification in order to foster the progress in this area. As such, the authors propose a framework that can serve as a formal common foundation for the convergence of various methodologies to assess diversity and novelty. Besides generalizing the existing metrics, the framework introduces rank sensitivity and relevance awareness as new two features in the measurement of novelty and diversity. In order to demonstrate the effects of the proposed metrics, the authors have conducted a set of experiments with different metric configurations and several baseline recommenders. Greedy diversification strategies are used to optimize for novelty and diversity.

The approach proposed by Belem *et al.* [BELÉM et al. 2013] provides a user with diverse and relevant recommendations of how to tag an object, formulating the task as a ranking problem. They propose a diversification method based on xQuad with a new formulation of the objective function. The diversification strategy is based on greedy heuristics, which always make the choice that looks best at the moment, but do not always guarantee the global optimum solution.

Other recent works addressing diversity in recommender systems include the contribution of Shi et al. [Shi et al. 2012b] that focuses on adapting the diversification level of the recommendation list to the individual needs of the target user, and the work of Hurley [Hurley 2013] proposes a method for maintaining high relevance of recommendations while incorporating the diversification criterion into a personalized ranking-based objective. In comparison to these approaches, our work does not focus on the personalized adaptation of item diversity for each user. We take a collaborative perspective on recommendation, aggregating preference rules for the overall users' behavior, and diversify w.r.t the current resource visited by the user.

5.8 Summary

In this chapter, we tackle the problem of generating user recommendations in an open-Web, cross-domain setting. We introduce a recommendation framework, whose novelty lies in the use of structural semantic information embedded in the content of the Web pages, and its combination with patterns of user browsing logs. In our cross-domain recommendation approach, we presented a method for predicting relevant resources to users by capturing the relational structure inherent in the content, and introduced a trade-off scheme between resource relevance and diversity.

In cross-domain recommender systems, the expectation is that the generated recommendations may be less precise than those provided when considering only one domain. The advantage may not be the improved accuracy, but the added novelty and diversity that may offer users higher satisfaction and utility.

However, this work presented an approach that is able to provide not only relevant recommendations, but also to control the trade-off between accuracy and diversity in order to keep relevance uncompromised. Through evaluation experiments on real-world datasets of semantically-enriched users logs, we showed the effectiveness of our approach and its superiority towards other popular hybrid recommender systems.

CHAPTER 6

Adaptive Cross-domain Collaborative Filtering with Probabilistic First-order Knowledge Transfer

In real-world Web recommender systems, users express partial preference feedback by rating only limited number of objects, which results in highly sparse user-object relations. To deal with such a bottleneck, we propose a novel cross-domain recommendation technique based on an adaptive approach. The contribution of this work is twofold. First, we present an expressive probabilistic first-order model to capture rich relational information in heterogeneous domains and reason about the relations through effective inference techniques. Second, we provide a mechanism for transferring knowledge from a source domain to a target domain in order to alleviate data sparseness. The approach tackles the most challenging case when users and objects in the two domains are not identical and do not overlap. We experimentally verify the efficacy of our approach and demonstrate that it outperforms several single-domain and cross-domain approaches.

6.1 Introduction

Extensive work has been done in the field of recommender systems to make use of the enormous online information of user activities for inferring user preference relationships about various products, books, web pages, or other information, which we generically refer to as objects. In the recommendation task, we are interested in predicting how likely a user is interested in a particular object, given information about this user, the other users' historical behavior, and information about the objects.

The majority of recommender systems particularly exploit preferences of users for objects, which are usually expressed in the form of ratings. However, in the greater part of recommender systems, especially those positioned in an open Web setting, users provide only limited preference feedback. This leads to a very sparse rating matrix.

One mechanism that is recently studied in order to deal with such bottlenecks is to borrow useful knowledge from another domain. The category of techniques applying such an approach falls under the adaptive-based group of cross-domain recommender systems. By applying Transfer Learning - a subfield of Machine Learning - the research community has been able to extend collaborative filtering to cross-domain settings where there is sparseness of explicit user-object overlap between domains. Transfer learning aims at improving the learning and prediction task in one domain by exploiting knowledge transferred from other domains. This is also the theoretical foundation that we investigate in this work. We propose an approach for learning useful knowledge of user-object preferences in a source domain, then transfer this knowledge in a sparse, target domain in order to enable there accurate user recommendations. There is one particular deficiency of the related cross-domain recommenders that apply an adaptive learning approach: they boil down to analyzing the user-object rating matrix solely as tabular data.

Traditional approaches to the problem derive from classical algorithms in statistical pattern recognition and machine learning. The majority of the approaches assume a flat data representation for each object, and focus on a single dyadic relationship between the objects. In web usage analysis, for example, the information sources might include user access logs, the relationships between the web pages visited, reviews written by the user, meta-data on the site and additional information about the user. This information can be aggregated in an e-commerce setting, where we include customers' buying patterns to make predictions about future purchases. In the Web context, where there is often much more relational information available than a single user-item relationship, we need added modeling power to capture richer relational information.

Therefore, our goal in this work is to formalize user preferential behavior with a richer model, which allows us to reason about many different relations at the same time. It takes advantage of the recent progress in statistical relational learning (a.k.a. multi-relational data mining), which provides rich representations and efficient learning algorithms for non-i.i.d. data. We propose a first-order probabilistic model, which is based on a powerful formalism that combines logical and probabilistic reasoning. It makes possible to combine many different objects and relations into a comprehensive solution to the recommendation task.

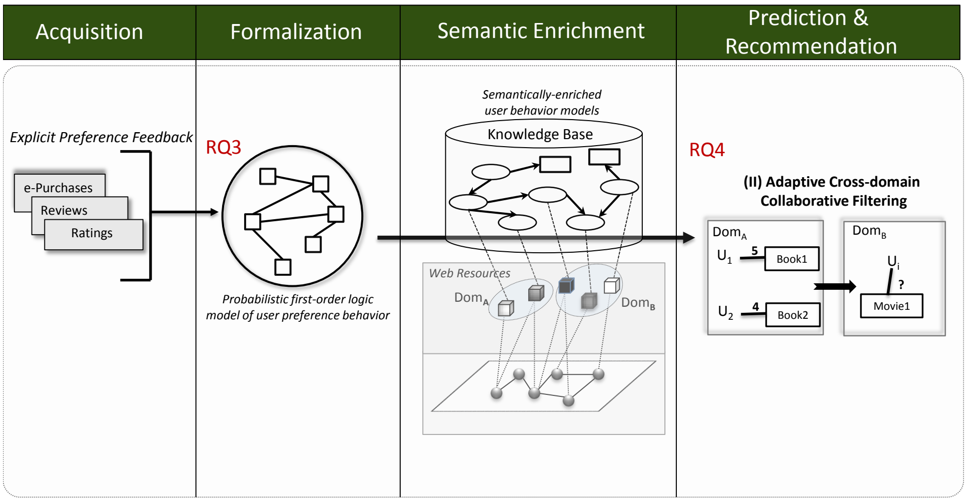

Acquisition
Formalization
Semantic Enrichment
Prediction & Recommendation
Explicit Preference Feedback
e-Purchases
Reviews
Ratings
RQ3
Probabilistic first-order logic model of user preference behavior
Semantically-enriched user behavior models
Knowledge Base
Web Resources
Dom$_A$
Dom$_B$
RQ4
(II) Adaptive Cross-domain Collaborative Filtering
Dom$_A$
U$_1$
5
Book1
U$_2$
4
Book2
Dom$_B$
U$_i$
?
Movie1

Figure 6.1: Framework overview: adaptive cross-domain collaborative filtering

In Figure 6.1, we graphically illustrate how the work presented in this chapter is positioned with respect to the overall thesis framework. The proposed approach captures the process starting with the acquisition of user preferential behavior in the form of explicit ratings, following with the probabilistic relational modeling of user behavior in each domain. In this stage, the meta-data harvested through the semantic enrichment strategies described in Chapter 4 can be used to leverage the description of objects. The formal model we provide perfectly accomodates the representation of such knowledge, since it is based on an expressive logic formalism. In the next step, the modeled preferential behavior is used as basis for the transfer approach, which selects knowledge from the source domain and applies it for preference predictions in the target domain.

6.1.1 Research Questions and Contributions

There are two main research questions that we address in this chapter, each of them also marking the contributions of our work:

Research Question 3. *Can we build a rich relational model of user behavior that can be used to accurately infer explicit user preferences to make recommendations?*

This research question addresses the challenge of finding an appropriate way of formalizing domain knowledge (or *domain theory*), so that we are able to capture multiple relations between objects and user preferences. The challenge here is to formalize the *domain theory* in a way that it provides user preferences representation, collaborative filtering features, multiple relations and attributes of users and objects/items, and rigorous formulation of uncertainty.

Research Question 4. *When user preference data for resources is very sparse, especially in an open Web setting, how can we transfer user behavior knowledge from one domain to better predict user preferences at a sparse target domain?*

This question addresses the data sparseness challenge. Our goal is to transfer knowledge from an auxiliary domain rich in training examples to a target domain which is highly sparse in user preferential feedback. At the same time, we need to provide a scalable learning and effective prediction approach.

The investigation of these research questions led to the following contributions:

Contribution III. *Probabilistic first-order model for hybrid recommendations.*

We present an expressive multi-relational model that makes it possible to combine many different objects and relations into a comprehensive solution to the recommendation task. We deploy a hybrid approach for generating recommendations, based on a content/collaborative merging scheme through feature combination.

Contribution IV. *Adaptive cross-domain collaborative filtering with probabilistic first-order knowledge transfer.*

We extend the expressive relational model of user-object preferences, provided in Contribution III, to build a new technique for knowledge transfer from one source domain to another sparse target domain. We contribute with a mechanism for generating accurate recommendations to users in a target domain that is unknown to them.

6.1.2 Outline

We refer in Section 6.2 to a set of works related to ours. We then proceed with our problem statement in Section 6.3. In Section 6.4, we introduce the first-order probabilistic model for formalizing domain knowledge and capturing user preferential behavior. In Section 6.5, we introduce the overall framework to learn useful knowledge in a source domain and transfer it to a sparse, target domain to enable accurate user preference predictions. Details on the inference mechanism and transfer process are given in Section 6.6 and Section 6.7, accordingly.

We have performed various experiments to evaluate the formalization and recommendation approach both in single domain and cross-domain setting. The experimental setup and evaluation results are shown in Section 6.8. We draw conclusions in Section 6.9.

6.2 Related Work

The general recommendation problem is built on the user-item matrix R of U users and I items, where the element r_{ij} is the rating given by user u to item i. In the matrix, a large scale of ratings are typically missing. Thus the recommendation task is formalized to predict the missing values in the matrix. The techniques are divided into content-based methods [Mooney and Roy 2000] and collaborating filtering (CF) methods [Ma et al. 2007, Zhang and Koren 2007].

There has been a plethora of collaborative filtering approaches introduced in the recommender systems field, but the factorization-based method has been demonstrated as most successful in performing the recommendation task with large-scale datasets [AGARWAL and CHEN 2009, KOREN et al. 2009]. Matrix factorization (MF) techniques learn hidden features from the observed ratings in a user-item matrix, also referred to as latent features of users and items. These latent features are used as basis for predicting the unobserved ratings in the matrix. A competitive representative of one of the state-of-the-art works is the probabilistic matrix factorization (PMF) model [SALAKHUTDINOV and MNIH 2007]. PMF follows a probabilistic approach for factorization in a single domain.

There is another line of works based on relational learning to analyze the probabilistic constraints between the attributes of entities and relationships. Xu *et al.* [XU et al. 2010] extend the expressiveness of relational models by introducing for each entity (or object) an infinite dimensional latent variable as part of a Dirichlet process mixture model. In an earlier work, Getoor *et al.* [GETOOR and SAHAMI 1999] present a conceptual model that allows one to reason about many different relations in a domain based on probabilistic relational models (PRMs). Yet, this work remains conceptual in describing how PRMs can be applied to CF and its efficacy is not experimentally verified.

Recently, there have been several cross-domain CF approaches proposed for dealing with recommendation across domains. Probabilistic matrix factorization is extended from single domain to multiple domains, such as in the work of Zhang *et al.* [ZHANG et al. 2010]. The authors propose a CF learning model that identifies correlations of ratings in a latent factor space. Thereby, rating matrices from different domains are transformed into user and item latent factors, which are then used for recommendations across domains. However, the approach requires that the sets of users in the different domains are the same. Shi *et al.* [SHI et al. 2011b] propose another interesting approach based on a graphical model for improving cross-domain CF by connecting multiple domains via user-assigned tags. They extend a matrix factorization approach to collaborative filtering by exploiting the tags given by users as source of valuable information that links users and items across various domains. In both these cases, it is assumed that there is information shared in both domains, such as in the form of tags, or more explicitly the sets of users/items.

However, our focus is particularly set on related works that apply an adaptive approach without requiring domain bridges. Like ours, these approaches aim to learn useful knowledge in an auxiliary domain, transfer it to another sparse domain in order to make better predictions there. Transfer Learning is an active research field in Machine Learning, which aims to improve a particular learning task in a specific domain by exploiting knowledge transferred from other domains [PAN and YANG 2010]. Transfer learning methods have been applied in the field of recommender systems to improve collaborative filtering.

Li *et al.* [LI et al. 2009b] propose a method referred to as codebook transfer (CBT), which consists of first compressing the ratings in the user-item matrix of an auxiliary domain into a compact cluster-level rating pattern. This structure is the *codebook*, which is then expanded in another sparse, target domain leading to the reconstruction of the respective rating matrix. The authors extend the approach by means of a probabilistic model, presenting in Li *et al.* [LI et al. 2009d] a common model built from the ratings of all the domains that does not need a dense source matrix to learn the implicit cluster-level pattern.

6.3 Problem Statement

The research problem we study in this paper is the effective generation of recommendations in a domain that is highly sparse in user preferential feedback by applying knowledge learned and transferred from an auxiliary domain. We adhere to again the definition [WINOTO and TANG 2008] of a domain as *the set of similar items with the same characteristics that can be easily differentiated, e.g. movies, concerts, songs, news, books, etc.*[1]

In the following, we give the definition of our cross-domain recommendation task. Without loss of generality, we define the task when two domains are involved. We use the notation introduced in [CREMONESI et al. 2011].

[1] We use the term *item* and *resource* interchangebly.

Definition 13. (Adaptive Cross-domain Recommendation Task) *Let $\mathcal{U}_{\mathcal{A}}$ be the set of users and $\mathcal{I}_{\mathcal{A}}$ the set of items in domain $\mathcal{A}$, as well as $\mathcal{U}_{\mathcal{B}}$ the set of users and $\mathcal{I}_{\mathcal{B}}$ the sets of items in a domain $\mathcal{B}$ that is very sparse on user-item ratings. Our task is to make* separate recommendations *of items in $\mathcal{R}_{\mathcal{B}}$ to users in $\mathcal{U}_{\mathcal{B}}$, given information on users in $\mathcal{U}_{\mathcal{A}}$, items $\mathcal{R}_{\mathcal{A}}$, and the respective ratings. We assume that $\mathcal{U}_{\mathcal{A}} \cap \mathcal{U}_{\mathcal{B}} = \emptyset$ and $\mathcal{I}_{\mathcal{A}} \cap \mathcal{I}_{\mathcal{B}} = \emptyset$.*

Our approach tackles the most challenging case when there is no user overlap and no resource overlap among domains (i.e. each domain has its own separate users and items). The task consists in transferring rules that capture user preference patterns data from a dense auxiliary rating domain (e.g. a popular book rating website) to a sparse rating domain (e.g. a new movie rating website). The goal is to improve recommendations of one domain from knowledge learned in other domains and alleviate the sparsity problem.

We illustrate our reseach problem with an example in Figure 6.2. Suppose we are given information on users, objects, and their explicit ratings in a source domain D_S, which is rather dense in terms of the ratings that users have expressed. At the same time, we also deal with another separate domain D_T, referred to as the target domain, which is highly sparse in user ratings.

In each domain, each user has attributes, such as address and age, and expresses own preferential feedback on objects (in this case books) via ratings. For example, user U_1^S in the source domain has rated $Book_1$ with a score of 5. The book has as author another object, which is from the country US. Similar relations are also occuring for user U_2^T, who rates $Book_1$ with score 5. We could potentially learn a pattern in this domain, such as intuitively this would be similar to "users of the same age like the same books" or "users from the same country like the same books". More importantly, we could learn in this domain how *strong* is this pattern, i.e. learn a weight.

In the target domain, we also have information on users and items, but in this setting the ratings are very sparse. The task is now to predict which score user U_2^T would give for $Movie_2$. We want to consider the rating similarities of this user to the other users (i.e. collaborative-filtering features), as well as the attributes of the book and the attributes of the user (i.e. content-based features). We are also interested to consider relations of this user to other objects, e.g. the tags assigned and how they are similar to those of other users. Since we have very few ratings available here to learn meaningful patterns between users and objects, we rely on potentially useful knowledge that can be transferred from the source domain $\mathcal{S}$. Precisely, the recommendation task consists in predicting the probability of the existence of a relation r^{ij} between user u_i^T and object o_j^T (e.g. $rates(u_2^T, movie_2)$), and then choosing as recommendations the set of objects with the highest probability value.

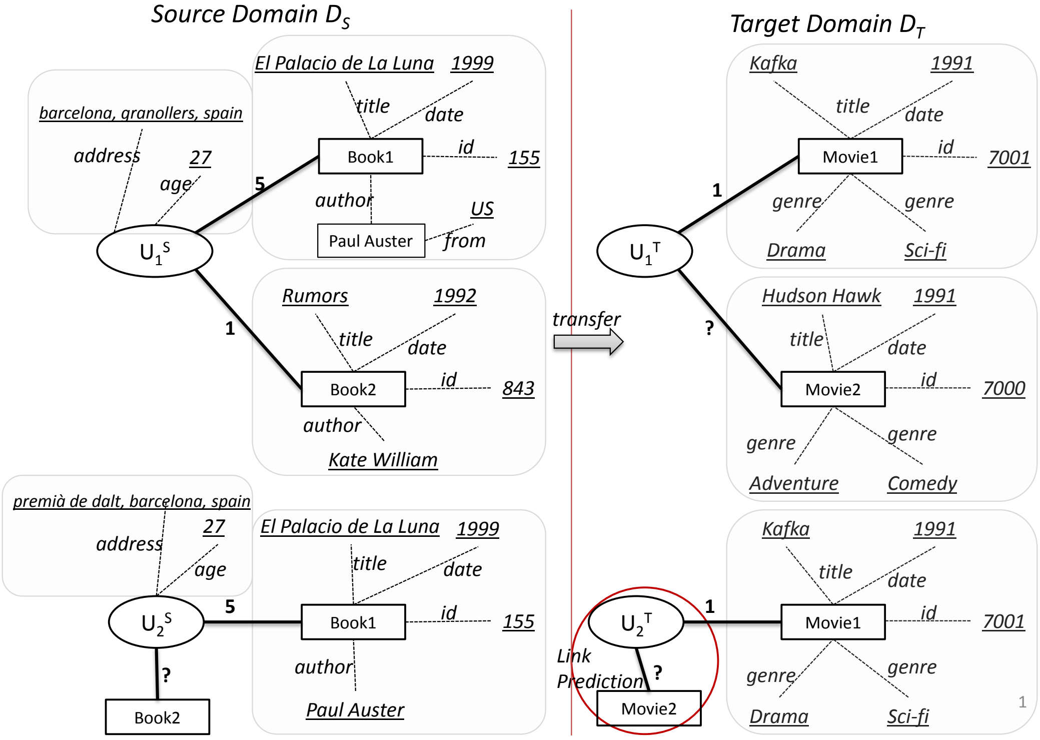


Figure 6.2: Example of cross-domain recommendation task in two domains

In our first step, we aim to expressively model the domain knowledge and respective relationships. This would allow us to consider more information and yield correct prediction values of the missing relations (in this case rating of user U_2^T for $Movie_2$). Therefore, we start by introducing a model for formalizing the theory i.e. objects, users, and relations in one domain. This is model is referred to as hMLN and is presented next in Section 6.4.

Since user and item profiles are distributed and do not overlap in these two domains, we have to establish a mechanism to learn meaningful knowledge about user preference ratings in one domain, and then transfer it to the other domain in a way that it enables us to make there accurate rating predictions. Section 6.5 presents the transfer approach that we propose for tackling this problem.

6.4 Domain Theory Formalization

Based on the given formulation of the prediction task, we introduce a model for representing the domain knowledge and the relationships between user and objects of the domain. As a representation formalism we use Markov logic, which generalises both first-order logic and probabilistic graphical models (Markov networks) by attaching weights to formulas in first-order logic. The weights in markov logic determine the degree to which the formulas they are attached to represent a constraint that is believed to hold. Markov logic is chosen for its generality and conceptual simplicity.

We deploy a hybrid approach that is based on a content/collaborative merging scheme through feature combination. Through the following model, we are able to express not only collaborative data, but also capture information on the inherent similarity of items that are otherwise opaque to a collaborative system. For simplicity of later references, we refer to this model as hMLN[2].

Conceptually, the model representing the domain theory consists of three parts:
(1) the MLN program, which itself contains two parts: (1.a) the predicate schema P and (1.b) the first-order logic formulae (rules) R,
(2) the set of evidences (examples), and
(3) the query set.

The evidence set, used for the training, is a list of ground atoms that are deemed to be true unless preceded by ”!”. The query set is the testing set, which consists of atoms whose arguments are variables. In our case, these define the relations we need to predict.

[2] hMLN standing from hybrid Markov Logic Network (MLN) model

(1.a) Predicate Schema	(2) Evidence	(3) Query
Rating(rating)	Rating(1)	rates("Bob", "book1", r)
User(pers)	Rating(2)	
Book(obj)	Rating(3)	
Author(pers)	Rating(4)	
Rating(rating)	Rating(5)	
Country(cntr)	User("Sara")	
hasAge(pers, age)	Book("book1")	
hasCountry(pers, cntr)	hasCountry("Sara","spain")	
hasAuthor(obj, author)	hasAge("Sara",32)	
hasRating(pers, obj, rating)	hasRating("Sara", "book1", 5)	
sameUser(pers, pers)		
sameCountry(cntr, cntr)	sameUser("Sara","Sara")	
sameBook(obj, obj)	User("Bob")	
sameAuthor(pers, pers)	hasAge("Bob",30)	
shareCountry(pers, pers)	hasCountry("Bob","spain")	
shareAuthor(obj, obj)	sameUser("Bob","Bob")	
shareAge(pers, pers)	shareCountry("Sara", "Bob")	
	shareAge("Sara", "Bob")	
	...	
(1.b) MLN Rules		
4.25	$\texttt{hasRating}(user_1, book_1, r) \wedge \texttt{shareAge}(user_1, user_2) \wedge \neg\texttt{sameUser}(user_1, user_2) => \texttt{rates}(user_2, book_1, r)$	
3.1	$\texttt{hasRating}(user_1, book_1, r) \wedge \texttt{shareAuthor}(book_1, book_2) \wedge \neg\texttt{sameBook}(book_1, book_2) => \texttt{rates}(user_1, book_2, r)$	
...		

Table 6.1: A sample of the proposed hMLN model. The goal is to find the highest probable rating ($r \in \{1, ..., 5\}$) that *Bob* gives to $book_1$. We define the schema as a list of predicate declarations. As evidence we are given profile information, as well as known ratings of *Bob* and other users. Any variable not explicitly quantified is universally quantified.

An example of domain theory formalized with hMLN is illustrated in Table 6.1. This an example from a book-rating domain. As we can see, the model contains four parts: predicate schema, evidence, rules and query. Next, we explain in more details the theory behind each part.

6.4.1 Predicate Schema

In order to model the objects and relations in a domain, we first need to define the predicate schema. The schema consists of a list of predicate declarations. Each predicate declaration specifies a predicate name with a list of argument types. Each type is supported by a set of constants. An argument may be either a variable or a constant. Variables start with a lower-case letter. A constant either 1) starts with a capital letter, 2) is a number, or 3) is a double-quoted string.
We distinguish between the following predicates:

- **Object-Declaration Predicate**
  $\texttt{Object}_{\texttt{i}}(o_i)$
  e.g. $\texttt{User}(person)$, $\texttt{Book}(book)$
  with evidence such as $\texttt{User}(``Anna")$, etc.

- **Object-Attributes Predicates**
  $\texttt{hasAttribute}_{\texttt{i}}(object_k, object_l)$
  or $\texttt{hasAttribute}_{\texttt{i}}(object_k, c)$, where c is a constant. The attribute can be a literal or another object.
  For example,
  $\texttt{hasAge}(person, age)$
  $\texttt{hasAuthor}(book, author)$
  $\texttt{hasCountry}(person, country)$

- **User-Object Preference Relations**
  $\texttt{hasRelation}_{\texttt{i}}(person, object, score)$
  For example: $\texttt{hasRating}(person, book, rating)$
  with an evidence like
  $\texttt{hasRating}(``Anna", ``book_2", 2)$.
  If the preference relation has a score other than binary, then a predicate with three arguments is defined. Instead of integer rating, we discretize levels of

preference scores, e.g. using the distribution:
L_1={0-2}, L_2={3-4}, L_3={5-6}, L_4={7-8}, L_5={9-10}.

Other examples of user-preference relations are: `tagged`, `visited`, `liked`, `purchased`, etc. A predicate definition preceded by "*" is considered as closed world assumption, i.e. all its ground atoms not listed in the evidence are false.

- **Recommendation Features Predicates**

  To model the dependency between objects we define a predicate, referred to as feature predicate:
  $\texttt{shareFeature}_{\texttt{i}}(object_k, object_l)$

  These features of dependencies may be qualitative or logical, which define if the relationship exists or not (e.g. $\texttt{sharePublicationDate}(book_k, book_l)$). We distinguish between `ObjectFeature` and `UserFeature` features, for example:
  $\texttt{shareAuthor}(\text{``}book_1\text{''}, \text{``}book_2\text{''})$: is a logical object feature, the arguments are instances of objects.
  $\texttt{shareCountry}(\text{``}user_1\text{''}, \text{``}user_2\text{''})$: is a logical user feature, the arguments are instances of users.
  $\texttt{shareAge}(\text{``}user_1\text{''}, \text{``}user_2\text{''})$: a qualitative user feature.

  In the case of the feature `shareAge`, we can also define intervals of t years of difference (e.g. $t = 5$) in order to group users based on their age.

- **Identity Predicates**

  $\texttt{sameObject}_{\texttt{i}}(object_k, object_k)$,
  $\texttt{sameBook}(book, book)$
  For example, $\texttt{sameUser}(person, person)$
  with an evidence $\texttt{sameUser}(\text{``}Anna\text{''}, \text{``}Anna\text{''})$.

  In general, the closed-world assumption is made, i.e. if the ground atoms are absent from the evidence set, then they are considered to be false. In cases when we have to check the value of the identity literals, such as $\texttt{sameBook}(book_1, book_2)$, we want to make sure that the ground atoms are included as evidence.

Evidence Set. In order to generate the evidence dataset, we populate the mentioned predicates with instantiations of the objects (i.e. information on users, books, and ratings).

6.4.2 Query Predicate

This is the preference relation, whose probability needs to be predicted. The score may be the rating value.
$\texttt{query\_relation}(person, object, score)$
For example, $\texttt{rates}(person, book, rating)$
with evidence $\texttt{rates}(\text{“}Bob\text{”}, \text{“}book_1\text{”}, L_5)$.

6.4.3 Hybrid Recommendation Formulae

A crucial part of MLNs is the set of formulae defined to model the dependencies between objects in the domain of interest. As explained earlier, the formulae (also referred to as *rules*) can be defined as hard or soft, and each has a particular weight. We define the following formulae:

Features Formulae: Modeled as hard rules, these formuale reflect the dependency of objects based on the attributes that they have in common.

Rule R.1

$$\texttt{hasAttribute}_\texttt{i}(o_1, a) \wedge \texttt{hasAttribute}_\texttt{i}(o_2, a)$$
$$\wedge\neg\ \texttt{sameObject}_\texttt{1}(o_1, o_2) => \texttt{shareFeature}_\texttt{i}(o_1, o_2).$$

For example, in our running scenario we would have the following rule to express the feature shareAuthor between any two objects of type Book that have the same Author in common:

$$\texttt{hasAuthor}(b_1, a_1)\ \wedge\ \texttt{hasAuthor}(b_2, a_1) \wedge$$
$$\wedge\neg\ \texttt{sameBook}(b_1, b_2) => \texttt{shareAuthor}(b_1, b_2).$$

Content-based Dependency Formulae: These are rules that express content-based dependencies between the score of the relation that we want to predict and the features of the objects. These are soft rules, whose weight we learn with parameter learning methods (Sec. 6.6.2).

Rule R.2

$$w_2 \ \texttt{hasRelation}_{\texttt{j}}(u, o_1, r) \wedge \texttt{shareObjectFeature}_{\texttt{i}}(o_1, o_2) \\ => \texttt{query\_relation}_{\texttt{j}}(u, o_2, r)$$

In our example, we would have the following rule:

$$4.8 \ \texttt{hasRating}(u, b_1, r) \wedge \texttt{shareAuthor}(b_1, b_2) => \texttt{rates}(u, b_2, r)$$

Collaborative-filtering Formulae: These rules reflect the similarity of behavior between user features and their rating behavior/preferences.

Rule R.3

$$w_3 \ \texttt{hasRelation}_{\texttt{j}}(u_1, o_1, r) \wedge \texttt{shareUserFeature}_{\texttt{i}}(u_1, u_2) \\ => \texttt{query\_relation}_{\texttt{j}}(u_2, o_1, r)$$

An example of this formula in our scenario would be the following rule, which implies that users of similar age rate the same book similarly:

$$4.25 \ \texttt{hasRating}(u_1, b, r) \wedge \texttt{shareAge}(u_1, u_2) => \texttt{rates}(u_2, b, r)$$

User-preference Dependency: These rules consider only the dependency on the similarity of user preference behavior w.r.t the relation that we want to predict.

Rule R.4

$$w_4 \ \texttt{hasRelation}_{\texttt{j}}(u_1, o_1, r_1) \wedge \texttt{hasRelation}_{\texttt{j}}(u_2, o_1, r_1) \wedge \\ \neg\texttt{sameObject}_{\texttt{k}}(u_1, u_2) \wedge \texttt{hasRelation}_{\texttt{j}}(u_1, o_2, r_2) \wedge \\ \neg\texttt{sameObject}_{\texttt{l}}(o_1, o_2) => \texttt{query\_relation}_{\texttt{j}}(u_2, o_2, r_2)$$

For example:

$$\begin{aligned} 2.78\ \ \mathsf{hasRating}(u_1, b_1, r_1)\ \wedge \mathsf{hasRating}(u_2, b_1, r_1) \wedge \\ \neg\mathsf{sameUser}(u_1, u_2)\ \wedge \mathsf{hasRating}(u_1, b_2, r_2) \wedge \\ \neg\mathsf{sameBook}(b_1, b_2) => \mathsf{rates}(u_2, b_2, r_2) \end{aligned}$$

6.5 METIS: Adaptive Cross-domain Recommendation Approach

The presented hMLN model is useful to formalize the knowledge of a particular domain. This is also referred to as the *domain theory*, which based on the theory explained above is composed of the following parts: predicate schema, rules, queries) and evidence (examples). This theory is formalized using the formulae presented earlier in Section 6.4.

Next, we introduce a mechanism to learn potentially useful knowledge from the available user preferences in a source domain, transfer this knowledge to a target domain that is sparse in such preferences, and make predictions in this target domain. A graphical illustration of the overall approach is displayed in Figure 6.3. The theory in domain $\mathcal{S}$ consists of the hMLN-based formulae comprising the set of predicates in the predicate schema, MLN rules, queries and evidence (i.e. ratings of users to objects). The same model is also used to formalize the domain theory in $\mathcal{D}$, which has only limited evidence available, that means there are only few examples of users and their ratings in this domain.

The procedure followed in our approach is presented in Algorithm 6. Given source domain theory and an initial target domain theory in $\mathcal{D}$, the algorithm outputs a revised and complete theory in $\mathcal{D}$, i.e. it generates answers for the queries in $\mathcal{D}$ which consists of user ratings predictions. In the first step, we learn the weights of the recommendation rules in $\mathcal{S}$ (Alg. 6, line 1), which is also referred to as the training phase where we use the known evidence available in the domain. In the following steps, we map the rules between the two domains, since they are expected to have different predicate schemas, and transfer the rules to domain $\mathcal{D}$.

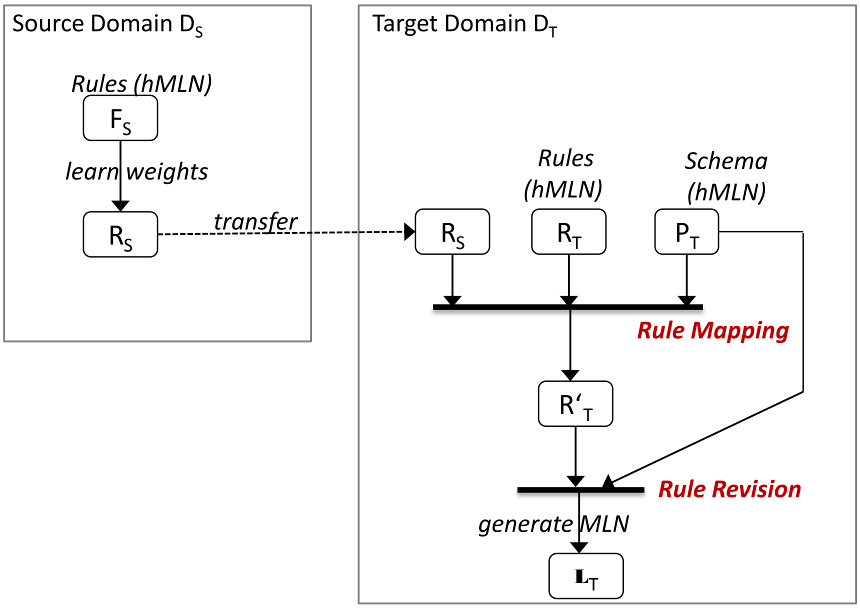


Figure 6.3: Cross-domain Knowledge Transfer

The rules are revised in $\mathcal{D}$, and the theory is accordingly completed by generating the MLNs and predictions in this domain.

Algorithm 6: Probabilistic First-order Knowledge Transfer

```
Input: Theory (predicate schema, rules, queries) and examples (evidence) in
    domain $\mathcal{S}$, initial theory in target domain $\mathcal{D}$
Output: Revised theory in domain $\mathcal{D}$
1: Rule weighting in domain $\mathcal{S}$
2: Rule mapping and transfer from domain $\mathcal{S}$ to domain $\mathcal{D}$
3: Rule revision in domain $\mathcal{D}$
4: Complete theory in domain $\mathcal{D}$
```

The task of generating predictions and learning weights of the rules is a probabilistic inference task. We explain next how we perform probabilistic inference in our approach. We then describe in details how rule mapping and rule revision is performed.

6.6 Probabilistic Inference

The task is to predict the probability of the query predicates, such as $rates(\text{"}Bob\text{"}, \text{"}book1\text{"}, r)$, given the evidence in the system. As such, we perform marginal inference in order to estimate the marginal probability of the atoms that consist our queries. The inference process consists of two main steps: grounding and then searching.

6.6.1 User Network Grounding

When generating the MLNs for large datasets, if we include all the users with whom a particular user u_i shares a relation (e.g. all the users who share same ratings with u_i), then the networks can become very large and inference is easily intractable. As such, we propose the following approach for network grounding in order to achieve better scalability during inference.

A network (the set of evidences, rules, and query formulae) is independently constructed for each user. This is what we referr to as *user graph* G_i, which is centered around a particular user u_i. Each *user graph* G_i contains information about the user u_i, the objects to which she has direct relationships, e.g. books rated. We also include in the graph a set K of "neighboring" users and their respective profile information.

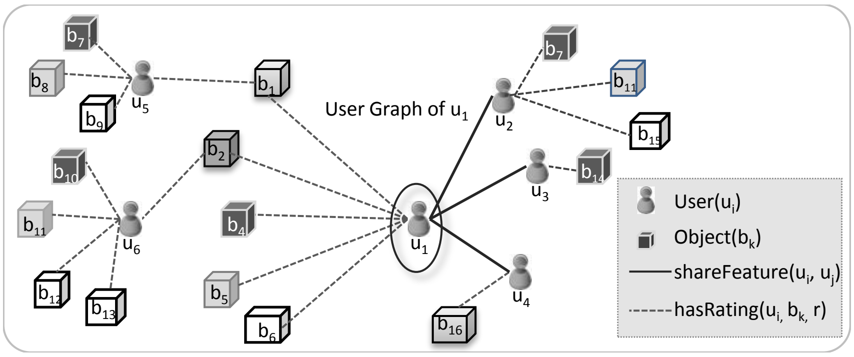


Figure 6.4: Individual *User Graph* used for grounding

An illustration of the user graph is depicted in Figure 6.4. The process of neighbor selection starts by filtering those users that share relations with u_i. For example, we would select in one group all the users who rated the same books with u_i, and in another group those users who share the same age with this user. Afterwards, the users in each group are ranked by a quantifiable measure of the relationship value they share with u_i. As such, the users in the first group would be ranked based on the number of co-rated books. Whereas, the neighbors in the other group would be ordered based on the close similarity of age they have with user u_i. Afterwards, the top-K set of neighbors is finally selected for each group. The default value of parameter K is 50, but it can also be tuned using training data. The last step is to include in the graph G_i all the objects to which every selected neighbor has direct relations (i.e. books rated).

Note that grounding is now performed for each graph independently. For each user we predict the relationship value (e.g. rating) to a set of query objects. We have in the end the set of all user graphs $G_i, G_u, ..., G_{|U|}$.

Grounding. For each user graph G, we fix its hMLN-based schema σ and domain of known constants C. Given the hMNL-based set of formula $\bar{F} = \{F_1, ..., F_N\}$ of G (in *clausal form* [3]) with weights $w_1, ..., w_N$, they define a probability distribution over *possible worlds*.

To construct this probability distribution, the first step is **grounding**: given a formula F with free variables $\bar{x} = (x_1, ..., x_m)$, for each constant $\bar{c} \in C^m$ we create a new formula $g_{\bar{c}}$, called a *ground clause*, which denotes the result of substituting each variable x_i of F with c_i. This process is performed for each formula F_i (for $i = 1...N$), where each ground clause g of F_i is assigned the same weight w_i.

The set of obtained ground clauses of $\bar{F}$ corresponds to a hypergraph where each atom is a node and each clause is a hyperedge. This graph structure is a Markov network as introduced in Section 2.2.2.

In a Markov network, for any possible world (instance) I, a ground clause g is *violated* if $w(g) > 0$ and g is false in I, or if $w(g) < 0$ and g is true in I. We denote the set of ground clauses violated in a world I as $V(I)$.

[3] Clausal form is a disjunction of positive or negative literals: e.g., the rule $R(a) => R(b)$ is equivalent to $\neg R(a) \vee R(b)$, which is in clausal form

The cost of the world I is

$$cost(I) = \sum_{g \in V(I)} |w(g)| \tag{6.1}$$

A lowest cost world I is called a *most likely world*. In order to find the most likely world or estimate the marginal probabilities of its atoms, we need to perform inference over the grounded network for each user graph.

Marginal Inference.

In our approach, we are interested in computing the highest probabilities for the queries posed as part of the relation prediction task. This consists in estimating the marginal probability of the query atoms, which is the process of marginal inference. Inference in MLNs is often regarded as infeasible because of the scalability issues associated with them. Yet, current state-of-the-art implementations show remarkable progress in overcoming these restrictions. We deploy marginal inference based on the MC-SAT algorithm [Poon and Domingos 2006], which applies slice sampling to Markov logic in combination with satisfiability testing by calling a heuristic SAT sampler. We apply the inference algorithm as implemented in the MLN inference engine Tuffy [4], which is recently shown to outperform all other engines in quality and efficiency [Niu et al. 2011].

6.6.2 Weight Learning

In our approach, we learn the weights of the formula discriminatively (maximizing the conditional likelihood of the query predicates given the evidence ones). Weight learning takes as input a training dataset and an MLN program without weight, then tries to compute the optimal weights of the MLN rules by maximizing the likelihood of the training data.

We use Diagonal Newton discriminative learner [Lowd and Domingos 2007] as implemented in Tuffy. In our approach, we learn the weights for each user network separately, then use their mean for the formulae that compose our final set.

[4] http://hazy.cs.wisc.edu/hazy/tuffy/

6.7 Transfer Process

6.7.1 Rule Mapping

The set of predicates used to describe data in the source and target domains may be partially or completely distinct. The first task in the transfer process is to establish a mapping from predicates in the source domain to predicates in the target domain. At this stage, we do not revise the weights of the rules, but focus on their structure. We deploy a *global mapping* approach: establish a mapping for each source predicate to a target predicate and then use it to translate the entire source MLN. While specific techniques can be used to discover mappings automatically, we assume here that the global mappings are already given.

For clarity, we illustrate in Table 6.2 an example of the rule mapping between the source and target domain. In this example, we deal with a set of predicates in the source domains, namely `User`$(person)$, `Book`$(object)$, `Category`(cat), `shareCategory`$(object, object)$, and `shareAge`$(person, person)$. There are also three rules defined in this domain. In the target domain, we have the following predicates: `User`$(person)$, `Movie`$(object)$, `Genre`(gen), and `shareGenre`$(object, object)$. We have at our disposal a global mapping, which in this case maps the predicate `User` of the first domain to the predicate `User` of the second domain, `Book` to `Movie`, `Category` to `Genre`, and finally `shareCategory` to `shareGenre`.

When the three rules of the source domain are then transferred to the target domain, the predicate `shareAge` is evaluated as invalid. This is done because the predicate is missing in the global mapping and is not contained in the schema of the target domain. As such, the rule containing this predicate is also made invalid and not used in the target domain. For the other two rules, we accordingly replace the predicates and variables using the information in the global mapping.

Source domain:	Target domain:
User$(person)$	User$(person)$
Book$(object)$	Movie$(object)$
Category(cat)	Genre$(genre)$
shareCategory$(object, object)$	shareGenre$(object, object)$
shareAge$(person, person)$	

Source Rules:

(1) hasRating(u_1, b_1, r) $\wedge$ shareCategory(b_1, b_2) $=>$ rates(u_1, b_2, r)

(2) hasRating(u_1, b, r) $\wedge$ shareAge(u_1, u_2) $=>$ rates(u_2, b, r)

(3) hasRating(u_1, b_1, r_1) $\wedge$ hasRating(u_2, b_1, r_1) $\wedge$ hasRating(u_1, b_2, r_2) $\wedge$ $\neg$ sameUser(u_1, u_2) $\wedge$ $\neg$ sameBook(b_1, b_2) $=>$ rates(u_2, b_2, r_2)

Mapping:	
User$(person)$ $\rightarrow$ User$(person)$	$\surd$
Book$(object)$ $\rightarrow$ Movie$(object)$	$\surd$
Category(cat) $\rightarrow$ Genre$(genre)$	$\surd$
shareCategory$(object, object)$ $\rightarrow$ shareGenre$(object, object)$	$\surd$
shareAge$(person, person)$	$\times$
(1) hasRating(u_1, b_1, r) $\wedge$ shareGenre(b_1, b_2) $=>$ rates(u_1, b_2, r)	$\surd$
(2) hasRating(u_1, b, r) $\wedge$ shareAge(u_1, u_2) $=>$ rates(u_2, b, r)	$\times$
(3) hasRating(u_1, b_1, r_1) $\wedge$ hasRating(u_2, b_1, r_1) $\wedge$ hasRating(u_1, b_2, r_2) $\wedge$ $\neg$ sameUser(u_1, u_2) $\wedge$ $\neg$ sameMovie(b_1, b_2) $=>$ rates(u_2, b_2, r_2)	$\surd$

Table 6.2: Rule mapping example

6.7.2 Rule Revision

The second step of the knowledge transfer process is to revise the source rules in order to improve their fit to the target data. The revision procedure focuses particularly on learning appropriate weights of the rules in the target domain.

We base our approach on previous works [MIHALKOVA et al. 2007, PAES et al. 2005, RICHARDS and MOONEY 1995] of first-order theory revision. We introduce the basic idea behind theory revision: one can start with a domain theory that may be approximate and incomplete and then correct for inaccuracies and incompleteness by training on examples. If the domain theory is at least approximately correct, we can learn faster with it than without it. Ideally, this should result in more accurate theories.

The problem tackled in our case is that perhaps not all the source rules are useful in the target domain, and not all the target theory can be explained/learned from the available source rules. Our work aims to address the following questions:

- If not all the source rules are related to the target task, how do we select the most relevant subset from the source domain rules?
- If not all the theory of the target domain can be explained or learned from the source rules, how do we identify the subset from the target domain that can benefit the most from the knowledge transfer?

Adhering to the definitions of Paes et al. [PAES et al. 2005], we formulate our revision task as *generalization*, which involves improving the inferential capabilities of a given probabilistic first-order theory by adding previously missing answers. The revision approach starts from an initial theory that is then minimally modified to become consistent with a set of given examples. In our case, we deal with positive examples only. This initial theory is divided in two parts: (i) background knowledge, which is the predicate schema that is assumed to be correct, and (ii) the knowledge that can be modified by the revision, in our case the rules.

Definition 14. *(Revision State). A* `revision state` *is defined as a tuple* $(\mathcal{T}, \mathcal{R}, \mathcal{C}^+, \mathcal{F})$ *consisting of a fixed probabilistic first-order theory* $\mathcal{T}$*, the set of probabilistic first-order rules* $\mathcal{R}$*, a set of positive examples* $\mathcal{C}^+$*, and a probabilistic evaluation function* $\mathcal{F}$*.*

We introduce the notion of *consistency* of revision states to express the condition that the revised theory logicaly implies all the examples and maximizes a given evaluation function.

Definition 15. *(Revision State Consistency). A revision state is* `consistent` *and denoted as* $(\mathcal{T},\mathcal{R}^{\models},\mathcal{C}^{+},\mathcal{F})$ *if its background theory and rules logically imply all the examples* $\mathcal{T} \sqcup \mathcal{R}^{\models} \models \mathcal{C}^{+}$ *and maximize the probabilistic evaluation function* $\mathcal{F}$.

The theory in our case is a *Markov logic* program. The dataset $\mathcal{C}^{+}$ of examples consists of the literals obtained after grounding. Rule revision, presented in Algorithm 7, consists in using an initial probabilistic first-order theory consisting of fixed background knowledge $\mathcal{T}$, rules $\mathcal{R}$, a set of positive examples $\mathcal{C}^{+}$, and a probabilistic evaluation function $\mathcal{F}$, in order to find a consistent revision state $(\mathcal{T},\mathcal{R}^{\models},\mathcal{C}^{+},\mathcal{F})$. We achieve this by performing probabilistic revision of the theory in the target domain. In probabilistic revision, the current structure is retained and the probability distributions that maximize the given probabilistic evaluation function are searched, resulting in a consistent revision state (according to Definition 15). This process boils down to parameter revision of the theory, which in our case is the task of learning the weight of the rules.

Algorithm 7: Rule Revision Algorithm

```
Input:   Theory $\mathcal{T}$, rules $\mathcal{R}$, evidence $\mathcal{C}^{+}$ and evaluation function $\mathcal{F}$
Output:  $(\mathcal{T},\mathcal{R}^{\models},\mathcal{C}^{+},\mathcal{F})$ a consistent revision state
repeat
    for rule $\alpha_i \in \mathcal{R}$ do
        Perform probabilistic revision
    end for
until state $(\mathcal{T},\mathcal{R}^{\models},\mathcal{C}^{+},\mathcal{F})$ is consistent
```

There exists various algorithms that can be used for parameter learning in MLN. In our approach, we perform discriminative weight learning with the Diagonal Newton discriminative learner as presented in Lowd and Domingos [Lowd and Domingos 2007]. This is a gradient descent-style algorithm, which deploys a preconditioned scaled conjugate gradient. The discriminative training method minimizes the negative conditional likelihood of the query predicates given the evidence ones.

Thereby, the evaluation function $\mathcal{F}$ is the negative conditional log-likelihood ($NCLL$), which is defined as $NCLL(\mathcal{T}|B) = -CLL(\mathcal{T}|B)$ with CLL being the conditional log-likelihood function [FRIEDMAN et al. 1997]:

$$CLL = \sum_{i=1}^{n} \log P(y_i|x_{i,1}, ..., x_{i,v-1}) \tag{6.2}$$

where $B_i = \{y_i, x_{i,1}, ..., x_{i,v-1}\}$ and y_i represents the class in the example i.

Maximizing the conditional likelihood of the class is equivalent to minimizing the classification error. Conditional likelihood is preferable in classification problems, where a theory with the smallest classification error needs to be found.

For the rules transferred from the source domain, we keep their original weights if they are positive, otherwise assign a value of 1. After various trials, we witness that performing parameter revision with negative weights leads to intractable processes. Meanwhile, we assign the weight of value 1 to the rules of the target domain.

6.8 Experimental Results

Our experiments are organized in two parts. The first part consists of experiments conducted in the single-domain case, where we need to test the accuracy of our approach within one domain. The second part consists of the cross-domain case, where we evaluate the mechanism of transferring knowledge across domains and test the accuracy of predictions in the target domain.

6.8.1 Datasets

The experiments are conducted on the following three publicly available datasets:

- **MovieLens**[5]: The original MovieLens dataset contains 10 million ratings (1-5 scales) from 71576 users and 10681 movies. For a better comparison with existing approaches, we follow the evaluation procedure of Shi *et*

[5] http://www.grouplens.org/node/73

al. [SHI et al. 2013], by selecting a subset with the first 5000 users and 5000 movies according to the identifiers in the original dataset. In the following, this dataset is denoted as ML.

- **LibraryThing**[6]: The original LibraryThing dataset contains ca. 750 thousand ratings from 7279 users and 37232 books, and in the subset we also select the first 5000 users and 5000 books [CLEMENTS et al. 2010]. This dataset is denoted as LT. The statistics of the ML and LT datasets are summarized in Table 6.3.

- **BookCrossing**[7]: This is a dataset of ratings from an online book club where users can rate books. In prior work [ZIEGLER et al. 2005], book ratings were collected from this site[8].

  We use this dataset for one specific type of experiments to evaluate recomendation utility in the single-domain case. This dataset is also used in recent studies on information heterogeneity in recommender systems [CANTADOR et al. 2011]. We performed a cleanup of the data, since it is quite noisy: there are invalid ISBNs, and some of the ISBNs in the rating file cannot be found in the book description file. Statistics of this dataset, denoted as BX, are displayed in Table 6.4. We tests with various subsets by filtering users based on different numbers of minimal ratings.

	Nr. users	Nr. items	Nr. Ratings	Sparseness
ML	5000	5000	584628	97.70%
LT	5000	5000	179419	99.30%

Table 6.3: Statistics of the datasets ML and LT

As in the related work of Shi *et al.* [SHI et al. 2013], we follow the subset selection procedure rather than random selection, in order to ensure accurate performance comparison and future experimental reproducibility.

[6] http://ir.ii.uam.es/hetrec2011/datasets.html

[7] http://www.bookcrossing.com

[8] http://www.informatik.uni-freiburg.de/cziegler/BX/

Min. ratings	Nr. Users	Nr. Books	Nr. Ratings
5	5628	57,324	136,284
10	3056	52,528	119,563
30	1053	42,340	86,928
50	568	36,194	68,361

Table 6.4: Statistics of the BookCrossing dataset (BX)

6.8.2 Experimental Setup for Single-domain Case

Evaluation methods for recommender systems are manifold, comprising statistical techniques to measure deviations of predicted and actual rating values, and approaches to estimate the utility of the recommendation list for the active user, e.g., precision and recall known from information retrieval.

In order to provide a comprehensive evaluation, we perform experiments for both aspects. As such, we organize the experiments in two parts: one for the *recommendations utility evaluation*, and the other for the *error deviation evaluation*. We chose to conduct experiments in three different datasets, in order to show not only the feasibility, but also the empirical expressiveness of our model.

Experimental Protocol for Utility Evaluation

We use decision-support metrics to evaluate the effectiveness of assisting users to select high-quality items from the overall set of items. We intend to judge how *relevant* a set of ranked recommendations is for the active user, thus, follow a methodology that estimates the utility of recommendations. As in Ma *et al.* [MA et al. 2007], the first 50% of the ratings from each user are utilized for training, and the rest are utilized for testing.

At first, we select the users with at least 5 ratings ($min\_ratings = 5$). In addition, we perform other tests applying different values of $min\_ratings$, in order to see how the approach reacts to the cases when the users provide more preference judgments. We generate top-10 recommendations lists and perform 10-fold cross-validation. For this analysis we use the BX dataset.

Experimental Protocol for Error Deviation Evaluation

We follow the experimental procedure of Shi *et al.* [SHI et al. 2013], which even though focuses on the cross-domain recommendation task, offers extensive evaluations of a single domain case such as ours. The comparative analysis is performed on ML and LT datasets, where each is divided into training set (60%), test set (20%), and validation set (20%). For each user in the test set, a small set of ratings (denoted as UPL for user profile length) is held out and included in the training set. In our case, UPL is set to 5. The rest of the ratings is used for testing applying 10-fold cross-validation.

6.8.3 Experimental Setup for Cross-domain Case

We follow the experimental procedure of Shi *et al.* [SHI et al. 2013]. The comparative analysis is performed on ML and LT datasets, where each is divided into training set (60%), test set (20%), and validation set (20%). For each user in the test set, a small set of ratings (denoted as UPL for user profile length) is held out and included in the training set. The rest of the ratings is used for testing applying 10-fold cross-validation.

6.8.4 Evaluation Metrics

For the utility evaluation case, we use established metrics of precision and recall. The recall metric [SARWAR et al. 2000] finds the percentage of test set objects in the dataset T_o occurring in recommendation list R_o, with respect to the overall number of test set objects $|T_o|$:

$$Recall = 100 \cdot \frac{|T_o \cap R_o|}{|T_o|} \tag{6.3}$$

Precision represents the percentage of test set objects occurring in the recommendation list, with respect to the size of the recommendation list:

$$Precision = 100 \cdot \frac{|T_o \cap R_o|}{|R_o|} \tag{6.4}$$

In addition, we report on the F1 measure as a combined metric for precision and recall.

For the evaluation case of measuring error deviations, we use mean absolute error (MAE) as the standard evaluation metric [MA et al. 2007, SHI et al. 2013] for measuring recommendation performance on rating-based recommender domains:

$$MAE = \frac{\sum |r_{u,o} - \bar{r}_{u,o}|}{|T_o|} \tag{6.5}$$

where $r_{u,o}$ denotes the rating that user u gave to object o, and $\bar{r}_{u,o}$ denotes the rating that user u gave to object o which ispredicted by our approach, and $|T_o|$ denotes the number of tested ratings.

6.8.5 Parameter Setting

For weight learning, we use the following parameters: number of samples for MC-SAT is set to 50, and max. iterations to 100. For user network grounding, we set the neighbor clustering parameter(Sec. 6.6.1) to $K = 20$.

6.8.6 Results of the Single-domain Case

We concentrate on separate, single domains and compare the performance of the proposed METIS approach with a set of alternative recommendation approaches listed below:

- **User-based Collaborative Filtering (UBCF)** is a representative of memory-based CF approaches, being one of the most popular recommendation techniques, because of its simplicity and high quality of recommendations. We apply Pearson correlation for similarity values and set neighborhood to 50.

- **Item-based Collaborative Filtering (IBCF)** is chosen as another popular recommendation method. It is a representative of model-based CF methods [SARWAR et al. 2000], which in addition to CF considers the item-item similarities.

- **Probabilistic matrix factorization (PMF)** [SALAKHUTDINOV and MNIH 2007] is a state-of-the-art model-based CF approach. The regularization parameter is set to 0.01.

We summarize below the results of the comparative analysis and the observations regarding the hMNL approach.

Metrics	Min.Ratings=5		
	UBCF	IBCF	METIS
Recall	5.76	**7.32**	5.12
Precision	3.69	3.64	**4.67**
F1	4.49	4.86	**4.88**

Table 6.5: Recommendation performance on dataset BX

Results of Utility Evaluation

Results illustrated in Table 6.5 show that user-based CF and item-based CF exhibit almost the same accuracy, indicated by the precision values. Their difference in recall shows a behavior change with respect to the types of users used in the scoring. Our approach, yields a recall value similar to the user-based method, but outperformes the other methods in precision. This can be explained by the fact that our method is accurate in making predictions, but the restrictions we have put in the size of networks per user cause a decrease in the number of relevant items predicted. It means that our choice on achieving scalability comes as a trade-off with recall.

In terms of F1 measure, which combines both recall and precision, METIS outperforms the other approaches. It is also important to highlight that while recall is an important metric, it is borrowed from information retrieval and is more useful in that setting than in recommender systems. The reason is that in recommender systems we aim to get the best five or three, or even the top one most relevant resource, i.e. with highest predicted probability. Therefore, while it is important to consider recall in some scenarios and optimize it accordingly, in our case this is not very sensitive and as crucial as precision.

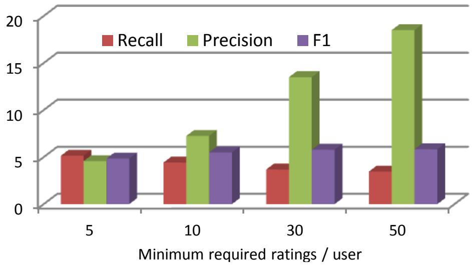


Figure 6.5: Performance for varying $min\_ratings$

In Figure 6.5, we observe that for increasing values of the minimum ratings applied to the evaluation setting, METIS provides much higher accuracy of the recommendation results. In particular, precision largely increases when we filter users that have 30-50 ratings. From the observations, we draw a conclusion that METIS is precise in making relevant recommendations, but its coverage is restricted to the scale of networks that we construct for inferencing. With larger networks recall can increase, but scalability needs to be always taken into consideration to ensure tractable solutions.

Results of Error Deviations Evaluation

Since IBCF is computationally expensive, we have used UBCF and PMF as better representative of state-of-the-art for this comparative analysis.

Table 6.6: Recommendation performance in the error deviation case between METIS and baseline approaches

Dataset(metric)	UBCF	PMF	METIS
ML(MAE)	0.833	0.831	**0.641**
LT(MAE)	0.857	0.771	**0.738**

As can be seen in Table 6.6, METIS outperforms the other approaches with regard to the mean absolute error in both datasets ML and LT. Our approach ensures high accuracy because of the power to handle the probability distributions of the closest dependencies that help in defining users' ratings.

6.8.7 Results of the Cross-domain Case

In this subsection, we compare the performance of the proposed METIS approach with a set of alternative recommendation approaches for the cross-domain case. As mentioned before, the performance is reported based on the test set with 10 randomly separated folds.
We compare our solution to the following approaches:

- **UBCF**: User-based collaborative filtering is used as a representative of memory-based CF approaches. The neighborhood size is tuned to 50.

- **PMF**: Probabilistic matrix factorization [Salakhutdinov and Mnih 2007] is one of the most successful model-based collaborative filtering approach. Note that the PMF and UBCF are designed for single domain recommendations.

- **CBT**: Codebook transfer [Li et al. 2009b] is a state-of-the-art cross-domain collaborative filtering approach. One domain (e.g., ML1) is used as the auxiliary domain to construct a codebook, and the other domain (e.g., LT) as the target domain in which the recommendations are generated. Following the experimental protocol used in [Li et al. 2009b, Shi et al. 2013], 500 users and 500 items with most rating are selected to construct the auxiliary domain. The number of clusters is set to 50 for both users and items.

- **RMGM**: Rating-matrix generative model [Li et al. 2009d] state-of-the-art cross-domain CF approach. The number of both the user and the item clusters is set to 20. It is to be highlighed that this approach is not purely adaptive, rather it builts a common model from the ratings of all the considered domains. Therefore, its comparison to our approach is to be particularly considered with a grain of salt.

In Table 6.7, we illustrate the results of performance comparison between METIS and other approaches. As mentioned previously, for these approaches we follow the protocol and results reported in the work of Shi *et al.* [Shi et al. 2013]. We draw the following observations. For the *LT-ML* case, the knowledge is transferred from

	UBCF	PMF	CBT	RMGM	METIS
ML-LT	0.857±0.009	0.771±0.009	0.729±0.010	0.745±0.010	1.329±0.019
LT-ML	0.833±0.009	0.831±0.010	0.792±0.009	0.780±0.010	0.770±0.070

Table 6.7: Comparison of cross-domain recommendation performance between METIS and other approach w.r.t MAE metric

LT (LibraryThing) domain to ML (Movielens) domain, then predictions are performed and evaluated in ML using the mean absolute error (MAE) metric. Lower errors indicate higher accuracy of predictions. In this case, METIS outperforms all the other approaches. It yields an error of 0.770, which is lower than the error produced by other baselines.

For the *ML-LT* case, the other approaches perfom better then METIS and give lower prediction errors. This leads to the observation that the proposed approach is also sensitive to the domain where it is applied. For example, the LT domain relies more on user preferences expressed within that domain, rather than on preference patterns of other domains. As such, exploiting more rating patterns within LT is more beneficial. Yet, these ratings have to be available first, which excludes the cold-start scenarios.

Thereby, there is one very important aspect to consider in this experiment. We intentionally performed training on very small datasets of examples. The reason was to investigate how able are we to learn and transfer useful knowledge from other domains when our target domain has extremely few data of explicit user preferences. As such, our model was trained on 10% of the available examples set, whereas other approaches are trained on much larger datasets. CBT and RMGM are trained on datasets 5 times larger, whereas UBCF and PMF on dataset 30x larger. Yet, we can observe that even in such situations, the approach is able to learn rules in a separate domain, which when transferred and exploited to the target domain are still useful for making accurate predictions.

To understand the impact of knowledge transfer in particular for the *ML-LT* case, we performed some additional tests to observe the type and weight of rules that influence prediction accuracy. Table 6.8 shows the results of the weights learned for a set of rules in LT (target domain). In this example, rule R1 and R2 are transferred from ML domain, whereas the rules R3-R5 are defined in the LT domain. The weights of all the rules are learned in LT. Negative weights are handled by noting that a clause with weight $w < 0$ is equivalent to its negation with weight $-w$, and the negation of a clause is the conjunction of the negations of all of its literals. The focus of this experiment is to observe which rules are weighted higher and interpret them accordingly. For example, one of the rules transferred from ML domain has a high weight, meaning that it satisfies many examples in LT domain, hence it is important to be applied in LT as well. The results are also helping us to observe the standard deviation, i.e. weights of some rules oscillating from positive to negative among user profiles.

RuleNr	Weight	Std.Dev
Rule R1	-1.678	±2.922
Rule R2	9.968	±0.825
Rule R3	-0.215	±0.506
Rule R4	0.6385	±0.991
Rule R5	-0.4012	± 0.562

Table 6.8: Weights of rules learned in LT and standard deviation among user profiles

Based on this observation, we further tested MAE with MLN programs that include different set of rules, e.g. filtering out rules that were weighted negative. As such, we performed experiments on the LT domain with the MLN of the following configurations: (i) only rules from LT trained on 100 users, (ii) only positive rules transferred from ML and trained in LT with just 50 users, (iii) rules from both LT and ML trained on 100 users. The results are shown in Figure 6.6.

We can observe that the choice of rules included in the MLN program is very important, that is why additional structure learning methods should be explored to leverage our approach. Still, a very interesting observation is the impact of the transferred knowledge. We see that with the rules transferred from ML, we get even better results i.e. smaller error, MAE is 1.32, than training only with LT rules on the same dataset of few available ratings (MAE is 1.86).

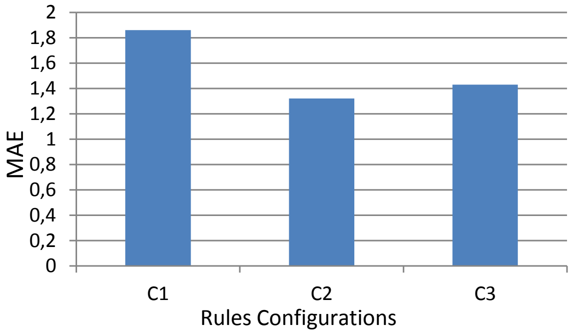


Figure 6.6: Experimental results with various configurations (set of rules). (C1) only rules from LT trained on 100 users, (C2) only positive rules transferred from ML and trained in LT with just 50 users, (C3) rules from both LT and ML trained on 100 users

6.8.8 Reproducibility of Experiments

We have conducted all our experiments with publicly available datasets. The approach uses implementations which are partly deployed in an existing, publicly platform (Tuffy), and the rest is made accessible online [9].

To facilitate further reproducibility of the experiments, we are also making available the MLN programs together with the designed schemas and query files for each of the datasets. The parameter setting used for the evaluations are explained above in Sec. 6.8.5 and Sec. 6.8.7.

6.9 Summary

In this section, we presented a novel adaptive approach, METIS, to address the cross-domain recommendation problem. There are two crucial components of this approach. The first component is the model introduced for capturing domain knowledge and the relations between users and object. To the best of our knowledge, this is the first work applying markov logic for the task of recommendation, particularly in a hybrid approach. The proposed theoretical model is generic and allows to model any domain of interest.

[9] http://www.github.com/jhoxha/metis/

The second important component is the mechanism for transferring first-order probabilistic knowledge learned in one domain to another target domain, where user preferences are very sparse. The goal is then to exploit the transferred knowledge in order to enable prediction of user preferences in the target domain. This approach is first of all helpful to comprehend the logically-expressed dependency rules, and therefore explain the reasoning behind the performed predictions. For example, we can gain better insights that dependencies on the age of the users play a greater role in predicting the rating behavior than the $country$ where the user come from.

Besides an increased comprehensibility, the approach is able to make accurate predictions by outperforming some of the most popular techniques in recommender systems. It achieves higher accuracy for the single-domain cases, and for particular settings of cross-domain cases. The significance of the approach lies in its ability to deal with very sparse information of user preferences. Even when trained on much smaller datasets of ratings than other state-of-the-art cross-domain CF methods, it is still able to make recommendations with similar or even smaller prediction errors than these methods.

Part III

Conclusions

CHAPTER 7

Conclusions and Outlook

We conclude this thesis by summing up the main hypothesis, the significance of the achieved results and conclusions drawn thereof. Finally, we provide an outlook on further research directions for future work.

7.1 Conclusions

The motivation for this thesis was driven by the latest developments on the Web, where people have to continously cope with large amount of information that is increasing daily. For humans, it is impossible to survey this information manually in order to find the relevant resources. Automated methods, such as recommender systems, are useful to facilitate the information seeking process by considering user preferential behavior expressed in terms of the navigation on the Web or explicit ratings. Yet, the information online is highly heterogeneous and the interest of the users span across different types of content and various domains. Based on this motivation, we raised the following main hypothesis in this thesis:

Accurate cross-domain user recommendations can be generated through predictive techniques, which leverage semantically-enriched models of user Web behavior.

The hypothesis was sustained with four research questions that were investigated and led to four significant contributions of the thesis.

The first contribution is an approach that formalizes user browsing behavior in an open Web setting. A significant result of this approach is an ontological model that can be reused by researchers and industry for user Web behavior modeling, eliminating ambiguity and facilitating data exchange. We also show that it is possible to enrich the information on user behavior by harvesting the knowledge available today on the Web in a structured form, which can also be easily processed by machines. We provide a set of techniques that enable such semantic enrichment. For the cases when such data is not available, we present a method to learn semantic types of Web resources via training with existing examples. This combination allows the overall approach to take advantages of both automated enrichment and machine learning solutions. The semantic formalization of user browsing behavior lays the foundation for effective techniques of behavior pattern analysis. We show that dynamic patterns of user browsing behavior can be discovered if we enable reasoning with semantic and temporal conditions.

The second contribution is a framework that is able to generate accurate recommendations of resources across domains. It exploits the implicit user preferences captured in the browsing behavior and the enriched semantics of Web resources harvested in the first contribution. This work is highly significant to present the value of adopting Semantic Web technologies in recommender systems that deal with heterogeneous information spread across different domains. The importance of our contribution lies in the stable, yet flexible methodology for establishing cross-domain bridges among the separate, diverse domains on the Web.

Our third contribution is an expressive theoretic model for making recommendations that allows us to reason about many different relations at the same time. Based on Markov logic, which is a simple and powerful language that combines first-order logic and probabilistic graphical models, the model is able to offer both expressivity and uncertainty handling. The proposed theoretical model goes beyond traditional ways of expressing user-object preference dependencies in a flat representation, focusing solely on dyadic relations. We can express various relationships and dependencies, and moreover, this is done through rules that are comprehensible to humans. This is of significant value if compared to other approaches that generally rely on unidentifiable clusters, or prediction models of the black-box type where it is difficult to explain the reasoning behind the recommender. Our model is generic and allows to capture various domains of interest. It also allows the flexibility of expressing both content-based and collaborative filtering features. Most importantly, we show that user preference prediction with our approach is very precise and outperforms traditional approaches with respect to accuracy measures.

The fourth contribution of our work is an adaptive cross-domain recommendation approach that deals with the cases when user preference data for resources is very sparse. In this setting, we consider explicit preference feedback of users, such as in the form of ratings. We extend the expressive relational model of user-object preferences, provided as part of the third contribution, to build a novel approach for knowledge transfer from one source domain to another sparse domain. The approach comprises a mechanism for generating accurate recommendations to users in a target domain that is unknown to them. We show that this approach is feasible, and based on various configurations (e.g. recommendation rules filtered out in the target domain) it leads to very accurate prediction of ratings. The significance of this contribution lies particularly in the ability to deal with extremely sparse datasets of user preferences, such as in cold-start scenarios where the content provider has very few or almost no knowledge on user behavior.

7.2 Outlook

The directions to continue the research presented in this thesis are manifold. In this section, we discuss some of these directions based on two main perspectives: technical and social.

I. Technical Perspective

Firstly, we discuss potential extensions of our work with respect to user browsing behavior modeling. In Section 4.2, we presented the WAM ontology to model user Web browsing behavior. We have captured high-level concepts regarding user profiles. The ontology can be easily extended to model additional data on users, such as demographic data, user identity, characteristics, capabilities, universal preferences, state of the user, application-specific preferences. Other temporal or dynamic features can also be included such as current activity, location, motion state and orientation. The formalism of our model facilitates its extension with existing user profile ontologies. Golemati *et al.* [GOLEMATI et al. 2007] present such an ontology that incorporates concepts and properties used to model user profiles. Razmerita *et al.* [RAZMERITA et al. 2003] present a generic ontology-based user modelling architecture applied to a Knowledge Management System. Heckman *et al.* [HECKMANN et al. 2005] present another top level ontology for user models referred to as GOMO, which also aims at facilitating exchange of user data and better interoperability for the applications in the Web. The captured user profile data can be embedded afterwards in behavior prediction and recommendation methods to enhance particularly the personalization of recommendations.

As part of WAM ontology, we model the concept wam:FunctionType to capture the type of function that a user Web browsing event serves (e.g. $booking$, $login$, $search$, buy, $listen$, etc.). We define this concept in the ontology because we anticipate its value in modeling the Web behavior of users. It is still to be inves-

tigated how we can harvest the instances of this class from the Web to annotate the resources navigated by users. In this work, we mainly focused on log semantic enrichment for the concept wam:ContentType, which relates to the content of resources the user is browsing (e.g. $movie$, $conference$, $song$, etc.). The semantic enrichment techniques can be further extended to discover the semantics of the function behind a browsing event. As such, we can later have more information regarding the type of browsed resource (e.g. song, book), and also know whether the song was listened or not, the book was bought or not by the user.
Regarding the typification of resources, our work lays the foundations for a promising learning problem where the output space is not a set of classes, but a structured, formal ontology containing also the relations among concepts. Machine learning techniques for structure prediction are the best candidates to be investigated in such problem.

The second aspect for future work relates to the collective cross-domain recommendation approach. Our work can be further extended with semantic-based features that can be engineered from the provided graph structure of the Web resources' content. This allows easy experimentation with other graph-based features besides the central set spreading feature we proposed. It is to be investigated which new features can leverage further the prediction performance.
Another direction for future investigation is the diversity maximization approach introduced in Section 5.5. In our work, we formulate one particular objective that is to be optimized. One can potentially formulate diversity and relevance as two different objectives, then the problem is to be tackled through dual objective optimizations techniques. The exploitation of those techniques to solve the constrained optimization problem of diversity enhancement and their impact on the overall recommendation performance can be a direction for future research.
An interesting approach would to extend the proposed framework with personalized recommendations that take into consideration short-term and long-term preferences of users. It can be useful to capture in different features a synthesized form of preferential behavior of a user in the past sessions (e.g. *Bob most often liked fiction movies*) and the preferences expressed in the current, ongoing session (e.g. *Bob is viewing comedy movies right now*). One has to investigate how to

distinguish between these two kinds of preferences with respect to different time segments or sessions, and study what role does each type play on improving the recommendation quality.

The third important avenue that requires further attention is the adaptive cross-domain recommendation approach. This is a very promising approach, which offers insightful and comprehensible recommendation rules while staying precise. We scratched the surface of that problem and raised attention to interesting developments that can follow up this work. It would be helpful to investigate how the scale of user graphs proposed for MLN construction influence the recommendation performance (w.r.t. recall). Another direction worth investigating is the structure learning problem for optimizing the selection of transferred rules from the source domain to the target domain.

Furthermore, handling and tuning of parameters in both adaptive and collective recommendation approaches requires substantial expertise. Future work should be concerned with the study of automated ways for parameter optimization.

II. Social Perspective

At last, we conclude with an important note related to a social aspect, namely the protection of user privacy. While recommender systems have gained great attention in the last decade, the issue of user privacy has constantly remained critical. Especially with the recent development of social media, personalization of recommendations and privacy preservation have been positioned in tension with each other. On behalf of increased recommendation accuracy and enhanced personalization, private companies in the industry are countinously gathering enourmous amount of personal user data, which also bear the risk of being used (or misused) otherwise for commercial purposes.

Even though the privacy issue has not been the focus of this work, we have tangentially addressed it in two different aspects. Firstly, the taxonomical representation in the WAM ontology, particularly with respect to the content type of the user browsing events, provide a way to abstract the information modeled about user behavior. For example, for a user visiting a page about *Lyon*, knowing that Lyon is a city in *France*, one we can use this higher abstraction of information (country instead of city) for the recommendation method, hiding in this case more specific details of what the user is viewing.

Furthermore, in the collective recommendation approach SUADEO presented in Chapter 5, we do not personalize the recommendations for each user. While the approach accomodates the case where user profile data can be used as features in the prediction technique, in our work we restrict to collaborative usage data that still ensure high accuracy. However, the presented recommendation framework can be extended in future work with privacy preserving techniques, which can exploit profile characteristics of the user for better personalization, yet ensure protection of their misuse. Some works that focus on establishing such balance propose different approaches that vary from proper agreement of user data sharing, to group-level recommendations, or trust-based knowledge sharing architectures. They can be starting points to further investigate how our work can be leveraged to ensure privacy of user data, for example modeling trust and featuring it in the prediction framework.

Appendix A: Publications

A.1 List of Publications

Julia Hoxha, Peter Mika, Roi Blanco. Learning Relevance of Resources across Domains to make Recommendations. *Proceedings of the 12th International Conference in Machine Learning and Applications* (ICMLA 2013), IEEE Computer Society, Miami, Florida, December 4–7, 2013.

Julia Hoxha, Achim Rettinger. First-order Probabilistic Model for Hybrid Recommendations. *Proceedings of the 12th International Conference in Machine Learning and Applications* (ICMLA 2013), IEEE Computer Society, Miami, Florida, December 4–7, 2013.

Julia Hoxha, Maria Maleshkova, Peter Korevaar. Knowledge Discovery meets Linked APIs. *International Workshop on Services and Applications over Linked APIs and Data (SALAD), at the 10-th Extended Semantic Web Conference* (ESWC 2013), CEUR Workshop Proceedings, pp. 56-65, Montpellier, France, May 26, 2013.

Armand Brahaj, Matthias Razum, Julia Hoxha. Defining Digital Libraries. Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013), Springer-Verlag, pp. 23-28, Valletta, Malta, September 22-26, 2013.

Julia Hoxha, Martin Junghans, Sudhir Agarwal. Enabling semantic analysis of user browsing patterns in the Web of Data. In *Proceedings of the 2nd International Workshop on Usage Analysis and the Web of Data (USEWOD), 21st International World Wide Web Conference* (WWW 2012), Arxiv abs/1204.2713, Lyon, France, April, 2012.

Andreas Scheuermann, Julia Hoxha. Ontologies for Intelligent Provision of Logistics Services. In *Proceedings of the 7th International Conference on Internet and Web Applications and Services* (ICIW 2012), XPS, Germany, May, 2012.

Julia Hoxha, Sudhir Agarwal. Semantic Formalization of Cross-site User Browsing Behavior. In *Proceedings of the 2012 IEEE/WIC/ACM International Conference on Web*

Intelligence, WI 2012, pp. 488-495, IEEE Computer Society, Macau, China, December 4-7, 2012.

Armand Brahaj, Detlev Doherr, Julia Hoxha. Behavior-Based Information Seeking in Digital Libraries. In *Proceedings of The 2nd International Multi-Conference on Complexity, Informatics and Cybernetics* (IMCIC 2011), Springer, Orlando, Florida, Mach, 2011.

Julia Hoxha, Armand Brahaj, Denny Vrandecic. open.data.al - Increasing the Utilization of Government Data in Albania. In *Proceedings of the 7-th International Conference on Semantic Systems, Triplification Challenge*, ACM, pp. 237–240, Graz, Austria, September, 2011.

Julia Hoxha, Anisa Rula, Basil Ell. Towards Green Linked Data. In *Proceedings of the Second International Workshop on Consuming Linked Data (COLD2011)*, CEUR Workshop Proceedings (CEUR-WS.org), October, Bonn, Germany, 2011.

Julia Hoxha, Armand Brahaj. Open Government Data on the Web: A Semantic Approach. In *Proceedings of the 2-nd International Conference on on Emerging Intelligent Data and Web Technologies* (EIDWT-2011), Springer, Tirana, Albania, September, 2011.

Julia Hoxha, Sudhir Agarwal. Semi-automatic Acquisition of Semantic Descriptions of Processes in the Web. In *Huang, J. X., King, I., Raghavan, V. V., and Rueger, S. (eds.): Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence* (WI 2010), IEEE Computer Society, pp. 256–263, Toronto, Canada, August 31–September 3, 2010.

Julia Hoxha, Andreas Scheuerman, Stephan Bloehdorn. An Approach to Formal and Semantic Representation of Logistics Services. In *Kerstin Schill and Bernd Scholz-Reiter and Lutz Frommberger, Proceedings of the Workshop on Artificial Intelligence and Logistics (AILog) at the 19th European Conference on Artificial Intelligence* (ECAI 2010), pp. 73–78, Lisbon, Portugal, September, 2010.

Appendix B: Behavior Pattern Analysis

B.1 Querying User Browsing Patterns

Understanding user behavior in accessing Web resources is a powerful tool for Web site providers to improve their applications, the design and content of the Web sites, to analyze users' navigation intentions and, respectively, improve search or build adaptive sites.

A crucial aspect in analyzing browsing behavior is its *temporal* dynamics related to the order of requests being issued. We present a query formulation and answering approach that is able to search for expressive patterns in terms of temporal constraints[1]. We apply a formalism that extends description logic with temporal constructs (such as what happens next, eventually, always) based on Linear Temporal Logic [Lutz et al. 2008].

We further show how to search for behavioral patterns upon the semantically formalized usage logs applying the query answering technique. The adaptation and application of this logic and techniques for Web usage analysis are novel. This work was presented in the publication [Hoxha et al. 2012].

B.1.1 $\mathcal{DL}$-LTL Formalism

It is important to reason about the browsing behavior of users not only using conditions related to the enriched semantics, but also addressing temporal constraints regarding the dynamics of such behavior. To achieve this, we allow queries upon our knowledge base to be formulated involving temporal operators such as eventually, until, always.

[1] This is joint work with colleagues Anees Mehdi and Martin Junghans

This motivates us to address temporal logics capable of ontological reasoning. To support such temporal reasoning, we follow an approach similar to the one in [BAADER et al. 2008], which presents the temporalized description logic $\mathcal{ALC}$-LTL as an extension of $\mathcal{ALC}$ with Linear Temporal Logic (LTL).
Instead of $\mathcal{ALC}$, we apply in our approach $\mathcal{SHOIN}(\mathbf{D})$, the DL underlying OWL-DL, which is used to express our WAM ontology. We refer to this formalism as $\mathcal{DL}$-*LTL* and use it for formulation of queries upon the formalized browsing events and sessions.
As in $\mathcal{ALC}$-LTL$|_{gGCI}$, we treat the TBox axioms as global i.e., the semantics of GCIs shall be respected at every time point.
First, we define the way to represent our knowledge base as a transition system in order to allow for reasoning with DL and LTL conditions. This system conforms to a $\mathcal{DL}$-LTL -structure, which we define as follows:

Definition 16. *A* $\mathcal{DL}$-LTL structure *is a sequence* $\mathfrak{I} = (\mathcal{I}_i)_{i=0,1,\ldots}$ *of standard interpretations* $\mathcal{I}_i = (\Delta, \cdot^{\mathcal{I}})$ *such that each interpretation* $\mathcal{I}_i$ *satisfies the unique name assumption.*

The unique names assumption (UNA) is the assumption that distinct ground terms denote different individuals (see [BAADER et al. 2008] for details). In our case, different names always refer to different entities within one interpretation. Next, we explain how this $\mathcal{DL}$-LTL structure is used for querying with temporal and semantic constraints.

B.1.2 Query Formulation

In our querying approach, we consider the notions of real time and abstract time. As mentioned in the definition of a session (Def. 9) in Section 4.2, the ordering of events within the session is used as basis for abstract time to formulate temporal constraints. As an example, let's consider the following query:

> Query q_1: Find sessions where user starts browsing *homepage* of Web site w_a, then *eventually* performs *search* in an engine (site), afterwards returns back to site w_a.

Answering q_1 requires finding a session with a start time T_s and end time T_e, and also the set of events $e_i \in S$ s.t. $T_s \leq e_i.t \leq T_e$. Within this set of events, we filter only those events having the required URLs and order. The former part of the query deals with the real time, whereas the latter temporal part deals with the abstract time (ordering of events in a timeline).
This query can be expressed as a $\mathcal{DL}$-LTL formula, whose notion we define inductively as follows:

- every ABox assertion is a $\mathcal{DL}$-LTL formula. For example, $\texttt{Event}(e)$ and $\texttt{hasURL}(a, b)$ are $\mathcal{DL}$-LTL formulas.

- if φ and ψ are $\mathcal{DL}$-LTL formulas, then so are $\varphi \wedge \psi$, $\varphi \vee \psi$, $\neg\varphi$, $\varphi \texttt{U} \psi$, and $\texttt{X}\varphi$ ($\texttt{U}$ being the $Until$ operator and $\texttt{X}$ the $Next$ operator).

As an example, the $\mathcal{DL}$-LTL formula:

$$(\texttt{Event}(e_1) \wedge \exists\ \texttt{contentType.Publication}(e_1))\wedge$$
$$\texttt{X}(\texttt{Event}(e_2) \wedge \exists\texttt{contentType.Search}(e_2))$$

describes an event e_1 with content type *Publication* and followed by an event e_2 of event type *search engine*.

The *validity* of a $\mathcal{DL}$-LTL formula ψ in a given $\mathcal{DL}$-LTL structure $\mathfrak{I} = (\mathcal{I}_i)_{i=0,1,...}$ at time point $i \in \{0, 1, ...\}$ is inductively defined as follows:

$$\begin{array}{lll}
\mathfrak{I}, i \models C(a) & \textit{iff} & a^{\mathcal{I}_i} \in C^{\mathcal{I}_i} \\
\mathfrak{I}, i \models R(a, b) & \textit{iff} & (a^{\mathcal{I}_i}, b^{\mathcal{I}_i}) \in R^{\mathcal{I}_i} \\
\mathfrak{I}, i \models \varphi \wedge \psi & \textit{iff} & \mathfrak{I}, i \models \varphi \textit{ and } \mathfrak{I}, i \models \psi \\
\mathfrak{I}, i \models \varphi \vee \psi & \textit{iff} & \mathfrak{I}, i \models \varphi \textit{ or } \mathfrak{I}, i \models \psi \\
\mathfrak{I}, i \models \neg\varphi & \textit{iff} & \mathfrak{I}, i \not\models \varphi \\
\mathfrak{I}, i \models \texttt{X}\varphi & \textit{iff} & \mathfrak{I}, i+1 \models \varphi \\
\mathfrak{I}, i \models \varphi\texttt{U}\psi & \textit{iff} & \textit{there is } k \geq i \textit{ such that } \mathfrak{I}, k \models \psi \\
 & & \quad \textit{and } \mathfrak{I}, j \models \varphi \textit{ for all } j, i \leq j < k
\end{array}$$

We write $\mathfrak{I}, i \models \varphi$ to denote that φ is valid in $\mathfrak{I}$ at time point i, whereas $\mathfrak{I} \models \varphi$ to denote that φ holds in $\mathfrak{I}$ at all time points i.e. φ holds globally. As usual, we use $\texttt{true}$ as an abbreviation for $A(a) \vee \neg A(a)$, $\Diamond\varphi$ as an abbreviation for $\texttt{trueU}\varphi$ (diamond, which is read as "sometime in the future"), and $\Box\varphi$ as an abbreviation for $\neg\Diamond\neg\varphi$ (box, which is read as "always in the future").

For query answering, we need to check if a temporal pattern is satisfied within a session. By Definition 9, every session $S = (s, T_s, T_e, U)$ contains a sequence of events $s = \langle e_0, \ldots, e_n \rangle$ with $T_s \leq e.t \leq T_e$. Since the temporal operators refer to the ordering of the events in this sequence, we consider this order to be the timeline of the abstract time.

The semantics of $\mathcal{DL}$-LTL requires an infinite sequences of time points. As pointed out in ([Baier and Katoen 2008]), one can introduce the so-called *trap* *state*, i.e. instead of the sequence $e_0, \ldots, e_n$ we consider the infinite sequences $e_0, \ldots, e_n, e, e, e, \ldots$ for some event e. To check the satisfiability of a temporal pattern in a session, we define the notion of $\mathcal{DL}$-LTL structure corresponding to a session.

Definition 17. *Given an ontology $\mathcal{O}$ and a session $S = (s, T_s, T_e, U)$ with $s = \langle e_0, \ldots, e_n \rangle$, the $\mathcal{DL}$-LTL structure corresponding to S, denoted by $\mathfrak{I}(S)$, is a $\mathcal{DL}$-LTL structure $(\mathcal{I}_i)_{i=0,1,\ldots}$ with $\mathcal{I}_i = (\Delta, \cdot^{\mathcal{I}})$ such that*

- $\mathcal{I}_i \models \mathcal{O}$*, for each* $i = 0, 1, \ldots$ *, and*
- $\mathcal{I}_i \models e_i$*, for* $0 \leq i \leq n$

where $\mathcal{I}_i \models e_i$ means that all the assertions describing the event e_i are satisfied. Given a $\mathcal{DL}$-LTL -formula φ, we say φ is satisfied in S, written $S \models \varphi$, if and only if $\mathfrak{I}(S), 0 \models \varphi$.

As mentioned above, we are interested in queries that not only require checking the satisfaction of temporal patterns in a session, but also the satisfaction of certain conditions on the session itself. Similar to the description of an event, a session is also described with a set of ABox assertions.
We define an *atom over a given variable x* as an assertion parametrized with variable x, i.e., atoms are of the form $C(x)$, $R(x, a)$ or $R(a, x)$ for a concept C, a property R and a constant a. We define a query as follows:

Definition 18. *Let ω and φ be conjuctions of atoms over some variable x, s.t. ω denotes the conjuctions of atoms related to the session, and φ is a $\mathcal{DL}$-LTL formula representing the temporal pattern related to the events within the session.
A query Q over x is an expression of the following form*

$$Q(x) \leftarrow \omega, \varphi \qquad (1)$$

Given a set of sessions $\{S_1, \ldots, S_l\}$ along with an ontology $\mathcal{O}$, the answer $\texttt{Ans}(Q)$ to a query Q of form (1) over variable x is a subset of $\{S_1, \ldots, S_l\}$ such that for each $S \in \texttt{Ans}(Q)$ we have

- $\mathfrak{I}(S) \models \omega[x/S]$*, and*
- $\mathfrak{I}(S), 0 \models \varphi[x/S]$

where $\omega[x/S]$ ($\varphi[x/S]$) represents the conjunction of atoms obtained from $\omega(x)$ by replacing the variable x with S.

The answers to a query could be equivalently defined as "a session S is the set $\mathtt{Ans}(Q)$ for a query Q if and only if $\mathfrak{I}(S), 0 \models (\Box\psi \wedge \varphi)[x/S]$". We separate conditions (to be satisfied) on a session from the temporal patterns (to be verified) within the session in order to make the formulation of the query understandable.

Semantics of $\mathcal{DL}$-LTL formula
The semantics of a $\mathcal{DL}$-LTL formula φ with temporal constraints is defined over a finite-state transition system that represents the sequence of events within a session S. Let $F_S = (\mathcal{P}, p_0, \pi)$ denote a finite state automaton (FSA) representing the sequence of events in session S. The automaton is described by a final set $\mathcal{P}$ of states, a unique start state $p_0 \in \mathcal{P}$, and a transition function $\pi : \mathcal{P} \to \mathcal{P}$. Each state corresponds to a separate knowledge base (set of DL axioms of OWL ontology) that describes one event of the sequence, while the event sequence order is preserved by the transition function.
Below we give examples of three queries and their $\mathcal{DL}$-LTL formulation.
Query q_1: Find sessions where user starts browsing *homepage* of Web site w_a, then *eventually* performs *search* in an engine (site), aftewards returns back to site w_a.

$$\begin{aligned} q_1(s) \leftarrow\ & (\mathtt{Session}(s) \wedge \mathtt{hasEvent}(s, e_1) \\ & \wedge \mathtt{hasEvent}(s, e_n) \wedge \mathtt{hasEvent}(s, e_m)), \\ & (\mathtt{Event}(e_1)\ \wedge \mathtt{baseURL}(e_1, w_a) \wedge \\ & \exists \mathtt{functionType}.\mathtt{Homepage}(e_1)) \wedge \\ & \Diamond((\mathtt{Event}(e_m) \wedge \exists \mathtt{contentType}.\mathtt{EngineSearch}(e_m)) \wedge \\ & \mathtt{X}(\mathtt{Event}(e_n) \wedge \mathtt{baseURL}(e_n, w_a))) \end{aligned}$$

Query q_2: *Find sessions where user browses site related to* music*, then eventually performs a search in site* google.com *or site* yahoo.com*, then immediately comes back to a* page of a music group *in site w_b*

$$\begin{aligned} q_2(s) \leftarrow\ & (\mathtt{Session}(s) \wedge \mathtt{hasEvent}(s, e_1) \\ & \wedge \mathtt{hasEvent}(s, e_m)\ \wedge \mathtt{hasEvent}(s, e_n)), \\ & (\mathtt{Event}(e_1) \wedge \exists\ \mathtt{contentType}.\mathtt{Music}(e_1)) \wedge \\ & \Diamond((\mathtt{Event}(e_m) \exists \mathtt{contentType}.\mathtt{EngineSearch}(e_m)) \wedge \\ & \wedge (\mathtt{baseURL}(e_m, \text{“}\mathit{http://www.google.com}\text{”}) \\ & \vee \mathtt{baseURL}(e_m, \text{“}\mathit{http://www.yahoo.com}\text{”})) \wedge \\ & \mathtt{X}(\mathtt{Event}(e_n)\ \wedge\ \mathtt{baseURL}(e_n, \text{“}w_b\text{”}) \wedge \\ & \exists\ \mathtt{contentType}.\mathtt{MusicGroup}(e_n))) \end{aligned}$$

Query q_3: *Find sessions where user is browsing all the time sites w_a or w_b*

$$\begin{aligned} q_3(s) \leftarrow & (\texttt{Session}(s) \ \wedge \texttt{Event}(e) \ \wedge \texttt{hasEvent}(e)), \\ & \Box(\texttt{baseURL}(e, \text{“}w_a\text{”}) \ \vee \texttt{baseURL}(e, \text{“}w_b\text{”})) \end{aligned}$$

In the following section, we introduce the approach we apply for answering such queries.

B.1.3 Query Answering Technique

We present a query answering approach based on a matchmaking technique, which allows to automatically retrieve sessions of user browsing events that satisfy a set of semantic and temporal conditions. These conditions are formulated in a query q, which as defined in Def. 18 is an expression of the form $q(x) \leftarrow \omega, \varphi$, s.t. ω consists of constraints related to the session itself, and φ is a $\mathcal{DL}$-LTL formula representing the temporal pattern related to the events within the session. The query is posed upon the set of sessions $\mathcal{S}$, which are formalized with the approach described in Chapter 4 and that we have stored in a central repository. In order to find the answers to the query, we apply a matchmaking mechanism that proceeds as follows:

Step 1 - Checking Satisfiability of Session Constraints
This steps deals with expression ω of the query q, which defines constraints related solely to the attributes of the session. As explained earlier, ω consists of a conjunction of parametrized ABox assertions α_s. In this step, we check for each of the sessions in the repository if it satisfies the assertions, using HermiT reasoner [2].
For example, in this step we would retrieve sessions with starting time in July and $userIP$ of a particular value IP.

Step 2 - Checking Satisfiability of Temporal Constraints
The remaining part of the query Q consists of the $\mathcal{DL}$-LTL formula φ, which defines the temporal constraints over the events contained in the sessions that we retrieved in Step 1. Therefore, only the resulting set of sessions from step 1 (denoted as S^1) are now considered in Step 2.
In order to check the satisfaction of temporal constraints, we iterate over the sessions in S^1 and (a) build a finite state automaton (FSA) for each $S_i \in S^1$, afterwards (b) iterate over the states of the automaton in order to determine whether a condition holds in the respective state. In the sequel, we provide more details about this verification technique, which applies a model checking mechanism based on the modal μ-calculus [Kozen 1983]. The goal is to check if

[2] http://hermit-reasoner.com/

the formula q holds in the session $S_i \in S^1$ represented as a sequence of events.

Step 2(a) - Construction of the FSA

Based on the previously defined semantics of a $\mathcal{DL}$-LTL formula φ, we have $F_{S_i} = (\mathcal{P}, p_0, \pi)$ denote an FSA representing the sequence of events in session S_i. The automaton is described by a final set $\mathcal{P}$ of states, a unique start state $p_0 \in \mathcal{P}$, and a transition function $\pi : \mathcal{P} \rightarrow \mathcal{P}$.

We construct a F_{S_i} from each sequence of events in the session: the start state $p_0 \in \mathcal{P}$ of F_{S_i} is generated by adding the axioms of WAM ontology O_A that describes the domain terminology used for events (TBox) as well as ABox assertions related to the session itself and its first event (with order 1).

The subsequent states and transitions are generated according to the event sequence in S_i. For a sequence $\langle e_i, e_{i+1} \rangle$ of two subsequent events e_i and e_{i+1}, a transition $\pi = (e_i, e_{i+1})$ is added from the state p_i to a new state p_{i+1}. State p_{i+1} is created by adding the description of the event e_{i+1} and the static domain knowledge from O_A.

Step 2(b) - Verification of $\mathcal{DL}$-LTL formula in query q

For a given automaton F_{S_i} and a given the $\mathcal{DL}$-LTL formula φ, the matchmaker findes the subset of states of F_{S_i} in which the formula is satisfied. In case φ is defined as a composite formula, then it is broken down into a set of atomic formulas ϕ_i. The final result is aggregated from the results of the atomic formulas recursively according to the semantics of the query formalism. We proceed as follows for each of the atomic formulas:
If $\phi = \top$ (true), then all the states of F_S are returned.
If $\phi = \bot$ (false), then an empty set is returned.
If $\phi = \Psi$ (proposition Ψ over a desired event description) then according to the semantics of the query formalism, we need to find those states in which the proposition Ψ holds, i.e., the desired event occurred.

Iterating over all the states of the automaton F_S, we add a state p in the resulting set, if the proposition Ψ holds in state p. Since a state is an OWL ontology and a proposition is a data query[3], we execute the data query on the OWL ontology. If the result set of the data query is non-empty, then the proposition holds in the state, otherwise not. The end result of this verification technique is a set of sessions (URIs), in which the sequence of events satisifies the $\mathcal{DL}$-LTL formula part of the query q. A detailed explanation of the algorithm is presented in [AGARWAL et al. 2009].

[3] Currently, we only support conjunctive queries on ontologies with our own implementation of query evaluation based on HermiT reasoner.

B.1.4 Experimental Evaluation

We performed experiments to test the feasibility of the query answering technique. The datasets of usage logs from SWDF and DBpedia presented in Section 4.6.1 were used. We formulated and executed a set of $\mathcal{DL}$-LTL queries on the repository of formalized events. These queries are illustrated in Table B.1. We used an IBM Thinkpad T60 dual core, with 2 GHz per core, Windows 7 (32-bit) as the operating system, and a total of 2 GB memory.

Query	Description
Q_1	find all sessions starting with a search in a search engine, then followed by a browsing event in DBpedia
Q_2	find all sessions starting with a browsing event in SWDF, followed by a search in google.com
Q_2	find all sessions where users have visited pages of Tango Musicians in DBpedia
Q_4	find all sessions where users have browsed papers of the conference WWW2009 in SWDF, and then the page of Madrid in DBpedia
Q_5	find all sessions where the users have eventually visited English Artists in DBPedia

Table B.1: Set of queries used for usage analysis experiments

In order to check the performance and scalability of the query answering mechanism, we have performed tests with different number of sessions, which is a portion of those saved in the repository. We observed that the answering time varies slightly for the queries ($\sim$ 0.15 seconds), starting with 0.5 seconds for query execution in 500 sessions. For the largest number of sessions (up to 1000) upon which we performed tests, the answering time is still feasible and always below 1.4 seconds.

In the diagram of Figure B.1, we show query answering time, reporting separately the time it takes for the OWL reasoning and for the model checking when testing it for one query (Q_1) and different number of sessions. We observe that model checking time is minimal, and reasoning takes nearly 94% of the overall answering time.

Overall, we observe that the query answering mechanism performs within feasible times (taking into consideration that we have not applied indexing, optimization techniques, parallelization, etc.).

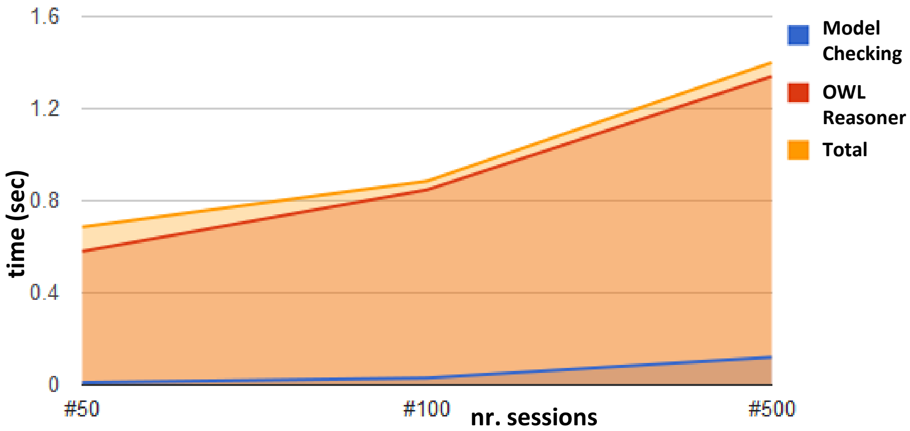


Figure B.1: Query answering time (composed of OWL reasoning time and model checking time) for one query and varying number of session

B.1.5 Summary

Additional dynamic aspects of user browsing behavior can be discovered if reasoning not only with semantic constraints, but also with temporal conditions is enabled. For this purpose, we introduce an approach to formulate queries using a temporalized description logic called DL-LTL, which combines $\mathcal{SHOIN}(\mathbf{D})$ with Lineal Temporal Logic. Alongside the formalism, we present a query answering mechanism, which is based on a model checking technique . This allows to automatically retrieve sessions of user browsing events that satisfy a set of semantic and temporal conditions.

We show the feasibility of our approach through experiments with usage logs from DBpedia and SWDF, providing in this way an exploratory analysis of the way users browse the Web of Data.

Bibliography

[ABEL et al. 2013] Abel, F., Herder, E., Houben, G.-J., Henze, N., and Krause, D. (2013). *Cross-system User Modeling and Personalization on the Social Web*. User Modeling and User-Adapted Interaction, 23(2-3), 169–209. (Cited on page 35.)

[ABRAMSON and AHA 2012] Abramson, M. and Aha, D. (2012). *What's in a URL? Genre Classification of Webpages from URLs*. In *The AAAI 2012 Workshop on Intelligent Techniques For Web Personalization and Recommender Systems (ITWP'12)*. (Cited on page 94.)

[ADDA et al. 2010] Adda, M., Valtchev, P., Missaoui, R., and Djeraba, C. (2010). *A Framework for Mining Meaningful Usage Patterns Within a Semantically Enhanced Web Portal*. In *Proceedings of the Third C* Conference on Computer Science and Software Engineering*, C3S2E '10, pp. 138–147, New York, NY, USA: ACM. (Cited on pages 35 and 94.)

[ADIDA and BIRBECK 2008] Adida, B. and Birbeck, M. (2008). *RDFa primer - bridging the human and data webs - W3C recommendation*. `http://www.w3.org/TR/xhtml-rdfa-primer/`. (Cited on page 70.)

[AGARWAL and CHEN 2009] Agarwal, D. and Chen, B.-C. (2009). *Regression-based latent factor models*. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pp. 19–28, New York, NY, USA: ACM. (Cited on page 142.)

[AGARWAL et al. 2009] Agarwal, S., Lamparter, S., and Studer, R. (2009). *Making Web services tradable - A policy-based approach for specifying preferences on Web service properties*. Web Semantics: Science, Services and Agents on the World Wide Web, Special Issue on Policies, 7, 11–20. (Cited on page 191.)

Bibliography

[AGRAWAL et al. 2009] Agrawal, R., Gollapudi, S., Halverson, A., and Ieong, S. (2009). *Diversifying search results*. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pp. 5–14, New York, NY, USA: ACM. (Cited on page 113.)

[ANAND and MOBASHER 2007] Anand, S. S. and Mobasher, B. (2007). *Contextual Recommendation*. In Berendt, B., Hotho, A., Mladenic, D., and Semeraro, G. (eds.): *From Web to Social Web: Discovering and Deploying User and Content Profiles*, pp. 142–160. Berlin, Heidelberg: Springer-Verlag. (Cited on pages 35 and 38.)

[AZAK 2010] Azak, M. (2010). *CrosSing A framework to develop knowledge based recommenders in cross domains*. In *MSc thesis, Middle East Technical University*. (Cited on page 40.)

[BAADER et al. 2008] Baader, F., Ghilardi, S., and Lutz, C. (2008). *LTL over Description Logic Axioms*. ACM Transactions on Computational Logic. (Cited on page 186.)

[BAADER and NUTT 2003] Baader, F. and Nutt, W. (2003). *The Description Logic Handbook*. In Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F. (eds.): *The Description Logic Handbook: Theory, Implementation, and Applications*, pp. 43–95: Cambridge University Press. (Cited on page 18.)

[BAEZA-YATES and RIBEIRO-NETO 2011] Baeza-Yates, R. A. and Ribeiro-Neto, B. A. (2011). *Modern Information Retrieval - the concepts and technology behind search, Second edition*: Pearson Education Ltd., Harlow, England. (Cited on page 33.)

[BAIER and KATOEN 2008] Baier, C. and Katoen, J.-P. (2008). *Principles of Model Checking*: The MIT Press. (Cited on page 187.)

[BAYKAN et al. 2009] Baykan, E., Henzinger, M., Marian, L., and Weber, I. (2009). *Purely URL-based Topic Classification*. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pp. 1109–1110, New York, NY, USA: ACM. (Cited on pages 94 and 95.)

[BAYKAN et al. 2011] Baykan, E., Henzinger, M., Marian, L., and Weber, I. (2011). *A Comprehensive Study of Features and Algorithms for URL-Based Topic Classification*. TWEB, 5(3), 15. (Cited on pages 77, 94 and 95.)

[BECHHOFER et al. 2004] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., and Stein, L. A. (2004). *OWL Web Ontology Language*. W3C Recommendation, W3C. `http://www.w3.org/TR/owl-ref` (accessed March, 2014). (Cited on pages 18 and 21.)

[BELÉM et al. 2013] Belém, F., Santos, R., Almeida, J., and Gonçalves, M. (2013). *Topic Diversity in Tag Recommendation*. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pp. 141–148, New York, NY, USA: ACM. (Cited on pages 134 and 135.)

[BERENDT et al. 2012] Berendt, B., Hollink, L., Hollink, V., Luczak-Rösch, M., Möller, K., and Vallet, D. (2012). *USEWOD2012 - 2nd International Workshop on Usage Analysis and the Web of Data*. In *Proceedings of the 21st International World Wide Web Conference (WWW2012), Lyon, France*. (Cited on pages 9 and 82.)

[BERENDT et al. 2003] Berendt, B., Mobasher, B., Nakagawa, M., and Spiliopoulou, M. (2003). *The Impact of Site Structure and User Environment on Session Reconstruction in Web Usage Analysis*. In *WEBKDD 2002 - Mining Web Data for Discovering Usage Patterns and Profiles*, vol. 2703 of *Lecture Notes in Computer Science*, pp. 159–179: Springer Berlin Heidelberg. (Cited on page 63.)

[BERKOVSKY et al. 2008] Berkovsky, S., Kuflik, T., and Ricci, F. (2008). *Mediation of User Models for Enhanced Personalization in Recommender Systems*. User Modeling and User-Adapted Interaction, 18(3), 245–286. (Cited on page 35.)

[BIN 2011] Bin, L. (2011). *Cross-domain collaborative filtering: A brief survey*. In *Procedings of IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 1085–1086. (Cited on page 40.)

[BIZER et al. 2013] Bizer, C., Eckert, K., Meusel, R., Mühleisen, H., Schuhmacher, M., and Völker, J. (2013). *Deployment of RDFa, Microdata, and Microformats on the Web - A Quantitative Analysis*. In *International Semantic Web Conference (2)*, vol. 8219 of *Lecture Notes in Computer Science*, pp. 17–32: Springer. (Cited on page 70.)

[BIZER et al. 2009] Bizer, C., Heath, T., and Berners-Lee, T. (2009). *Linked Data - The Story So Far*. International Journal on Semantic Web and Information Systems, 5(3), 1–22. (Cited on pages 66 and 72.)

Bibliography

[BOBADILLA et al. 2013] Bobadilla, J., Ortega, F., Hernando, A., and GutiéRrez, A. (2013). *Recommender Systems Survey*. Knowledge-Based Systems, 46, 109–132. (Cited on page 34.)

[BOSER et al. 1992] Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). *A Training Algorithm for Optimal Margin Classifiers*. In Haussler, D. (ed.): *Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92)*, pp. 144–152, Pittsburgh, PA, USA: ACM Press. (Cited on page 26.)

[BUCKLIN and SISMEIRO 2003] Bucklin, R. E. and Sismeiro, C. (2003). *A Model of Web Site Browsing Behavior Estimated on Clickstream Data*. Journal of Marketing Research, XL, 249–267. (Cited on pages 63 and 92.)

[BURGES et al. 2005] Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., and Hullender, G. (2005). *Learning to Rank Using Gradient Descent*. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, pp. 89–96, New York, NY, USA: ACM. (Cited on pages 122 and 123.)

[BURKE 2002] Burke, R. (2002). *Hybrid Recommender Systems: Survey and Experiments*. User Modeling and User-Adapted Interaction, 12(4), 331–370. (Cited on page 37.)

[CANTADOR et al. 2008a] Cantador, I., Bellogín, A., and Castells, P. (2008a). *A multilayer ontology-based hybrid recommendation model*. AI Commun., 21(2-3), 203–210. (Cited on pages 35 and 38.)

[CANTADOR et al. 2011] Cantador, I., Brusilovsky, P., and Kuflik, T. (2011). *2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011)*. In *Proceedings of the 5th ACM conference on Recommender systems*, RecSys 2011, New York, NY, USA: ACM. (Cited on page 162.)

[CANTADOR et al. 2008b] Cantador, I., Castells, P., and Superior, E. P. (2008b). *Extracting Multilayered Semantic Communities of Interest from Ontology-based User Profiles: Application to Group Modelling and Hybrid Recommendations*. In *Computers in Human Behavior, special issue on Advances of Knowledge Management and the Semantic*: Elsevier. In press. (Cited on pages 35 and 38.)

[CARBONELL and GOLDSTEIN 1998] Carbonell, J. and Goldstein, J. (1998). *The use of MMR, diversity-based reranking for reordering documents and producing summaries*. In *Proceedings of the 21st international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pp. 335–336, New York, NY, USA: ACM. (Cited on page 114.)

[CATLEDGE and PITKOW 1995] Catledge, L. D. and Pitkow, J. E. (1995). *Characterizing Browsing Strategies in the World-Wide Web*. In *Computer Networks and ISDN Systems*, pp. 1065–1073. (Cited on page 92.)

[CHAPELLE et al. 2009] Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P. (2009). *Expected reciprocal rank for graded relevance*. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pp. 621–630, New York, NY, USA: ACM. (Cited on pages 122 and 123.)

[CLEMENTS et al. 2010] Clements, M., de Vries, A. P., and Reinders, M. J. T. (2010). *The influence of personalization on tag query length in social media search*. Information Process Management. (Cited on page 162.)

[CODINA and CECCARONI 2010] Codina, V. and Ceccaroni, L. (2010). *Taking Advantage of Semantics in Recommendation Systems*. In *Proceedings of the 13th International Conference of the Catalan Association for Artificial Intelligence*, pp. 163–172, Amsterdam, The Netherlands, The Netherlands: IOS Press. (Cited on pages 35 and 37.)

[CODINA and CECCARONI 2012] Codina, V. and Ceccaroni, L. (2012). *Semantically-Enhanced Recommenders*. In Riaño, D., Onaindia, E., and Cazorla, M. (eds.): *Artificial Intelligence Research and Development - Proceedings of the 15th International Conference of the Catalan Association for Artificial Intelligence, University of Alacant, Spain, October 24-26, 2012*, vol. 248 of *Frontiers in Artificial Intelligence and Applications*, pp. 69–78: IOS Press. (Cited on pages 35 and 37.)

[CRAMMER et al. 2001] Crammer, K., Singer, Y., Cristianini, N., Shawe-taylor, J., and Williamson, B. (2001). *On the algorithmic implementation of multiclass kernel-based vector machines*. Journal of Machine Learning Research, 2, 2001. (Cited on page 76.)

[CREMONESI et al. 2011] Cremonesi, P., Tripodi, A., and Turrin, R. (2011). *Cross-Domain Recommender Systems*. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pp. 496–503, Washington, DC, USA: IEEE Computer Society. (Cited on pages 35, 40, 41, 100, 104, 133 and 143.)

[D'AQUIN et al. 2011] d'Aquin, M., L., S. E., and Motta, E. (2011). *Semantic Technologies to Support the User-Centric Analysis of Activity Data*. In *Workshop on Social Data on the Web Workshop (SDoW) at International Semantic Web Conference*. (Cited on page 93.)

[DOWNEY et al. 2007] Downey, D., Dumais, S. T., and Horvitz, E. (2007). *Models of searching and browsing: Languages, studies, and application*. In *Proceedings of IJCAI*, pp. 2740–2747. (Cited on page 92.)

[DOWNEY et al. 2008] Downey, D., Dumais, S., Liebling, D., and Horvitz, E. (2008). *Understanding the Relationship Between Searchers' Queries and Information Goals*. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pp. 449–458, New York, NY, USA: ACM. (Cited on page 92.)

[EIRINAKI et al. 2006] Eirinaki, M., Mavroeidis, D., Tsatsaronis, G., and Vazirgiannis, M. (2006). *Introducing Semantics in Web Personalization: The Role of Ontologies*. In Ackermann, M., Berendt, B., Grobelnik, M., Hotho, A., Mladenič, D., Semeraro, G., Spiliopoulou, M., Stumme, G., Svátek, V., and Someren, M. (eds.): *Semantics, Web and Mining*, vol. 4289 of *Lecture Notes in Computer Science*, pp. 147–162: Springer Berlin Heidelberg. (Cited on pages 35 and 38.)

[FERNÁNDEZ-TOBÍAS et al. 2011] 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*, HetRec '11, pp. 25–32, New York, NY, USA: ACM. (Cited on pages 35, 42, 100, 113 and 133.)

[FERNÁNDEZ-TOBÍAS et al. 2012] 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*, CERI '12. (Cited on pages 40, 41, 104 and 133.)

[FRIEDMAN et al. 1997] Friedman, N., Geiger, D., and Goldszmidt, M. (1997). *Bayesian Network Classifiers*. Mach. Learn., 29(2-3), 131–163. (Cited on page 161.)

[GAO et al. 2013] Gao, S., Luo, H., Chen, D., Li, S., Gallinari, P., and Guo, J. (2013). *Cross-Domain Recommendation via Cluster-Level Latent Factor Model*. In Blockeel, H., Kersting, K., Nijssen, S., and Železný, F. (eds.): *Machine Learning and Knowledge Discovery in Databases*, vol. 8189 of *Lecture Notes in Computer Science*, pp. 161–176: Springer Berlin Heidelberg. (Cited on page 35.)

[GENESERETH and NILSSON 1987] Genesereth, M. R. and Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (Cited on page 14.)

[GETOOR and SAHAMI 1999] Getoor, L. and Sahami, M. (1999). *Using Probabilistic Relational Models for Collaborative Filtering*. In *In Workshop on Web Usage Analysis and User Profiling (WEBKDD'99*. (Cited on page 142.)

[GETOOR and TASKAR 2007] Getoor, L. and Taskar, B. (2007). *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*: The MIT Press. (Cited on page 24.)

[GOLEMATI et al. 2007] Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G., and Halatsis, C. (2007). *Creating an Ontology for the User Profile: Method and Applications*. In *In Proceedings of the First International Conference on Research Challenges in Information Science (RCIS*. (Cited on page 178.)

[GROSOF et al. 2003] Grosof, B. N., Horrocks, I., Volz, R., and Decker, S. (2003). *Description Logic Programs: Combining Logic Programs with Description Logic*. In *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pp. 48–57, New York, NY, USA: ACM. (Cited on pages 18 and 21.)

[GRUBER 1995] Gruber, T. R. (1995). *Toward principles for the design of ontologies used for knowledge sharing*. Int. J. Hum.-Comput. Stud., 43, 907–928. (Cited on page 14.)

[HAYES 2004] Hayes, P. J. (2004). *RDF Semantics*. W3C Recommendation, W3C. `http://www.w3.org/TR/rdf-mt/` (accessed Aug. 15, 2013). (Cited on page 23.)

[HECKMANN et al. 2005] Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., and Wilamowitz-Moellendorff, M. (2005). *Gumo – The General User Model Ontology*. In Ardissono, L., Brna, P., and Mitrovic, A. (eds.): *User Modeling 2005*, vol. 3538 of *Lecture Notes in Computer Science*, pp. 428–432: Springer Berlin Heidelberg. (Cited on page 178.)

[HOXHA and AGARWAL 2010] Hoxha, J. and Agarwal, S. (2010). *Semi-automatic Acquisition of Semantic Descriptions of Processes in the Web*. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2010, Toronto, Canada, August 31 - September 3, 2010*, pp. 256–263: IEEE Computer Society. (Cited on page 9.)

[HOXHA and AGARWAL 2012] Hoxha, J. and Agarwal, S. (2012). *Semantic Formalization of Cross-Site User Browsing Behavior*. In *Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence, WI 2012, Macau, China, December 4-7, 2012*, pp. 488–495: IEEE Computer Society. (Cited on page 9.)

[HOXHA et al. 2012] Hoxha, J., Junghans, M., and Agarwal, S. (2012). *Enabling Semantic Analysis of User Browsing Patterns in the Web of Data*. In *2nd International Workshop on Usage Analysis and the Web of Data (USEWOD), 21st International World Wide Web Conference (WWW2012), Lyon, France*, vol. CoRR, abs/1204.2713. (Cited on pages 9 and 185.)

[HOXHA et al. 2013] Hoxha, J., Mika, P., and Blanco, R. (2013). *Learning Relevance of Resources across Domains to make Recommendations*. In *12th International Conference on Machine Learning and Applications, ICMLA, Miami, FL, USA, December 4-7, 2012. Volume 1*. (Cited on page 9.)

[HOXHA et al. 2014a] Hoxha, J., Mika, P., and Blanco, R. (2014a). *Cross-domain Recommendations using Structured Data*. Journal of Knowledge and Information Systems (under review). (Cited on page 9.)

[HOXHA and RETTINGER 2013] Hoxha, J. and Rettinger, A. (2013). *First-order Probabilistic Model for Hybrid Recommendations*. In *12th International Conference on Machine Learning and Applications, ICMLA, Miami, FL, USA, December 4-7, 2012. Volume 1*. (Cited on page 10.)

[HOXHA et al. 2014b] Hoxha, J., Rettinger, A., and Biba, M. (2014b). *From Movies to Books: Cross-domain Collaborative Filtering with Probabilistic First-order Knowledge Transfer*. Semantic Web Journal (under review). (Cited on page 10.)

[HURLEY and ZHANG 2011] Hurley, N. and Zhang, M. (2011). *Novelty and Diversity in Top-N Recommendation – Analysis and Evaluation*. ACM Trans. Internet Technol., 10(4), 14:1–14:30. (Cited on pages 134 and 135.)

[HURLEY 2013] Hurley, N. J. (2013). *Personalised Ranking with Diversity*. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pp. 379–382, New York, NY, USA: ACM. (Cited on page 136.)

[JÄRVELIN and KEKÄLÄINEN 2002] Järvelin, K. and Kekäläinen, J. (2002). *Cumulated Gain-based Evaluation of IR Techniques*. ACM Trans. Inf. Syst., 20(4), 422–446. (Cited on page 122.)

[JIN et al. 2004] Jin, X., Zhou, Y., and Mobasher, B. (2004). *A Unified Approach to Personalization Based on Probabilistic Latent Semantic Models of Web Usage and Content*. In *Proceedings of the AAAI 2004 Workshop on Semantic Web Personalization (SWP'04)*. (Cited on page 35.)

[JOACHIMS et al. 2009] Joachims, T., Hofmann, T., Yue, Y., and Yu, C.-N. J. (2009). *Predicting structured objects with support vector machines*. Commun. ACM, 52(11), 97–104. (Cited on page 77.)

[JOHNSON et al. 2004] Johnson, E. J., Moe, W. W., Fader, P. S., Bellman, S., and Lohse, G. L. (2004). *On the Depth and Dynamics of Online Search Behavior*. Manage. Sci., 50, 299–308. (Cited on page 92.)

[KAMINSKAS and RICCI 2011] Kaminskas, M. and Ricci, F. (2011). *Location-adapted Music Recommendation Using Tags*. In *Proceedings of the 19th International Conference on User Modeling, Adaption, and Personalization*, UMAP'11, pp. 183–194, Berlin, Heidelberg: Springer-Verlag. (Cited on pages 35, 40 and 41.)

[KAN and THI 2005] Kan, M.-Y. and Thi, H. O. N. (2005). *Fast webpage classification using URL features*. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pp. 325–326, New York, NY, USA: ACM. (Cited on pages 77, 78 and 94.)

[KAUTZ et al. 1996] Kautz, H., Selman, B., and Jiang, Y. (1996). *A General Stochastic Approach to Solving Problems with Hard and Soft Constraints*. In *The Satisfiability Problem: Theory and Applications*, pp. 573–586: American Mathematical Society. (Cited on page 31.)

[KOPPULA et al. 2010] Koppula, H. S., Leela, K. P., Agarwal, A., Chitrapura, K. P., Garg, S., and Sasturkar, A. (2010). *Learning URL patterns for webpage de-duplication*. In *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010*, pp. 381–390. (Cited on page 94.)

[KOREN et al. 2009] Koren, Y., Bell, R., and Volinsky, C. (2009). *Matrix Factorization Techniques for Recommender Systems*. Computer, 42(8), 30–37. (Cited on page 142.)

[KOZEN 1983] Kozen, D. (1983). *Results on the Propositional mu-Calculus*. Theor. Comput. Sci., 27, 333–354. (Cited on page 190.)

[Li et al. 2009a] 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 Jont Conference on Artifical Intelligence*, IJCAI'09, pp. 2052–2057, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (Cited on page 35.)

[Li et al. 2009b] Li, B., Yang, Q., and Xue, X. (2009b). *Can Movies and Books Collaborate?: Cross-domain Collaborative Filtering for Sparsity Reduction*. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, IJCAI'09, pp. 2052–2057, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (Cited on pages 143 and 168.)

[Li et al. 2009c] Li, B., Yang, Q., and Xue, X. (2009c). *Transfer Learning for Collaborative Filtering via a Rating-matrix Generative Model*. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 617–624, New York, NY, USA: ACM. (Cited on page 35.)

[Li et al. 2009d] Li, B., Yang, Q., and Xue, X. (2009d). *Transfer Learning for Collaborative Filtering via a Rating-matrix Generative Model*. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 617–624, New York, NY, USA: ACM. (Cited on pages 143 and 168.)

[Liddell and Scott 1940] Liddell, H. and Scott, R. (1940). *A Greek-English Lexicon*. (Cited on page 4.)

[Lin et al. 2007] Lin, H.-T., Lin, C.-J., and Weng, R. C. (2007). *A note on Platt's probabilistic outputs for support vector machines*. Mach. Learn., 68(3), 267–276. (Cited on page 109.)

[Liu et al. 2007] Liu, P., Nie, G., and Chen, D. (2007). *Exploiting Semantic Descriptions of Products and User Profiles for Recommender Systems*. In *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2007, part of the IEEE Symposium Series on Computational Intelligence 2007, Honolulu, Hawaii, USA, 1-5 April 2007*, pp. 179–185: IEEE. (Cited on page 35.)

[Loizou 2009] Loizou, A. (2009). *How to recommend music to film buffs: enabling the provision of recommendations from multiple domains*. PhD thesis, University of Southampton. (Cited on pages 35, 42, 100 and 133.)

[Lops et al. 2009] Lops, P., de Gemmis, M., Semeraro, G., Musto, C., Narducci, F., and Bux, M. (2009). *A Semantic Content-Based Recommender System Integrating Folksonomies for Personalized Access*. In Castellano, G., Jain, L., and

Fanelli, A. (eds.): *Web Personalization in Intelligent Environments*, vol. 229 of *Studies in Computational Intelligence*, pp. 27–47: Springer Berlin / Heidelberg. (Cited on page 37.)

[LOWD and DOMINGOS 2007] Lowd, D. and Domingos, P. (2007). *Efficient Weight Learning for Markov Logic Networks*. PKDD. (Cited on pages 156 and 160.)

[LUTZ et al. 2008] Lutz, C., Wolter, F., and Zakharyaschev, M. (2008). *Temporal Description Logics: A Survey*. In *Proceedings of the Fourteenth International Symposium on Temporal Representation and Reasoning*: IEEE Computer Society Press. (Cited on page 185.)

[MA et al. 2007] 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*, SIGIR '07, pp. 39–46, New York, NY, USA: ACM. (Cited on pages 141, 163 and 165.)

[MABROUKEH and EZEIFE 2009] Mabroukeh, N. R. and Ezeife, C. I. (2009). *Using domain ontology for semantic web usage mining and next page prediction*. In *CIKM*, pp. 1677–1680. (Cited on page 93.)

[MABROUKEH and EZEIFE 2011] Mabroukeh, N. and Ezeife, C. I. (2011). *Ontology-based Web Recommendation from tags*. In *Data Engineering Workshops (ICDEW), 2011 IEEE 27th International Conference on*, pp. 206–211. (Cited on page 35.)

[MAIDEL et al. 2008] Maidel, V., Shoval, P., Shapira, B., and Taieb-Maimon, M. (2008). *Evaluation of an ontology-content based filtering method for a personalized newspaper*. In *Proceedings of the 2008 ACM Conference on Recommender systems*, RecSys '08, pp. 91–98, New York, NY, USA: ACM. (Cited on pages 35 and 37.)

[MANNING et al. 2010] Manning, C. D., Raghavan, P., and Schutze, H. (2010). *Introduction to information retrieval*. Information Retrieval, 13(2), 192–195. (Cited on page 111.)

[MANOLA and MILLER 2004] Manola, F. and Miller, E. (2004). *RDF Primer*. W3C Recommendation, W3C. `http://www.w3.org/TR/rdf-primer/` (accessed Aug. 15, 2013). (Cited on pages 22 and 23.)

[MIDDLETON et al. 2009] Middleton, S. E., Roure, D. D., and Shadbolt, N. R. (2009). *Ontology-Based Recommender Systems*. In Staab, S. and Rudi Studer, D. (eds.): *Handbook on Ontologies*, International Handbooks on Information Systems, pp. 779–796: Springer Berlin Heidelberg. (Cited on page 35.)

[MIHALKOVA et al. 2007] Mihalkova, L., Huynh, T., and Mooney, R. J. (2007). *Mapping and revising Markov logic networks for transfer learning*. In *Proceedings of the 22nd national conference on Artificial intelligence - Volume 1*, AAAI'07, pp. 608–614: AAAI Press. (Cited on page 159.)

[MIKA 2011] Mika, P. (2011). *Microformats and RDFa deployment across the Web*. `http://tripletalk.wordpress.com/2011/01/25/rdfa-deployment-across-the-web/`. (Cited on page 70.)

[MIKA and POTTER 2012] Mika, P. and Potter, H. (2012). *Metadata Statistics for a Large Web Corpus*. In *Proceedings of the Linked Data Workshop (LDOW) at the International World Wide Web Conference*. (Cited on page 70.)

[MITCHELL 1997] Mitchell, T. M. (1997). *Machine Learning*: McGraw-Hill. (Cited on page 23.)

[MOBASHER et al. 2003] Mobasher, B., Jin, X., and Zhou, Y. (2003). *Semantically Enhanced Collaborative Filtering on the Web*. In *Web Mining: From Web to Semantic Web, First European Web Mining Forum*, vol. 3209 of *Lecture Notes in Computer Science*, pp. 57–76: Springer. (Cited on pages 35, 36, 37, 93 and 129.)

[MONTGOMERY and FALOUTSOS 2001] Montgomery, A. L. and Faloutsos, C. (2001). *Identifying Web Browsing Trends and Patterns*. Computer, 34, 94–95. (Cited on page 92.)

[MOONEY and ROY 2000] Mooney, R. J. and Roy, L. (2000). *Content-based book recommending using learning for text categorization*. In *Proceedings of the fifth ACM conference on Digital libraries*, DL '00, pp. 195–204, New York, NY, USA: ACM. (Cited on page 141.)

[NAKATSUJI et al. 2010] Nakatsuji, M., Fujiwara, Y., Tanaka, A., Uchiyama, T., and Ishida, T. (2010). *Recommendations Over Domain Specific User Graphs*. In *Proceedings of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pp. 607–612, Amsterdam, The Netherlands, The Netherlands: IOS Press. (Cited on page 35.)

[NEWELL 1982] Newell, A. (1982). *The Knowledge Level*. Artificial Intelligence, 18(1), 87–127. (Cited on page 14.)

[NGUYEN et al. 2010] Nguyen, T. T. S., Lu, H. Y., and Lu, J. (2010). *Ontology-Style Web Usage Model for Semantic Web Applications*. In *International Conference on Intelligent Systems Design and Applications (ISDA)*, pp. 784–789. (Cited on page 35.)

[NIU et al. 2011] Niu, F., Ré, C., Doan, A., and Shavlik, J. (2011). *Tuffy: scaling up statistical inference in Markov logic networks using an RDBMS*. Proc. VLDB Endow., 4(6), 373–384. (Cited on page 156.)

[NOY et al. 2001] Noy, N. F., Sintek, M., Decker, S., Crubézy, M., Fergerson, R. W., and Musen, M. A. (2001). *Creating Semantic Web Contents with Protégé-2000*. IEEE Intelligent Systems, 16(2), 60–71. (Cited on page 22.)

[OBERLE et al. 2003] Oberle, D., Berendt, B., Hotho, A., and Gonzalez, J. (2003). *Conceptual User Tracking*. In *Proceedings of the 1st International Atlantic Web Intelligence Conference on Advances in Web Intelligence*, AWIC'03, pp. 155–164, Berlin, Heidelberg: Springer-Verlag. (Cited on page 93.)

[OWL WORKING GROUP 2009] OWL Working Group, W. (2009). *OWL 2 Web Ontology Language: Document Overview*: W3C Recommendation. `http://www.w3.org/TR/owl2-overview/` (accessed Apr. 10, 2014). (Cited on page 22.)

[PAES et al. 2005] Paes, A., Revoredo, K., Zaverucha, G., and Costa, V. (2005). *Probabilistic First-Order Theory Revision from Examples*. In Kramer, S. and Pfahringer, B. (eds.): *Inductive Logic Programming*, vol. 3625 of *Lecture Notes in Computer Science*, pp. 295–311: Springer Berlin Heidelberg. (Cited on page 159.)

[PAN 2009] Pan, J. Z. (2009). *Resource Description Framework*. In Staab, S. and Studer, R. (eds.): *Handbook on Ontologies*, International Handbooks on Information Systems: Springer, 2nd ed. (Cited on pages 22 and 23.)

[PAN and YANG 2010] Pan, S. J. and Yang, Q. (2010). *A Survey on Transfer Learning*. IEEE Trans. on Knowl. and Data Eng., 22(10), 1345–1359. (Cited on page 143.)

[PAN et al. 2011] 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 Twenty-Second International Joint Conference on Artificial*

Intelligence - Volume Volume Three, IJCAI'11, pp. 2318–2323: AAAI Press. (Cited on pages 35 and 40.)

[PAN et al. 2010] Pan, W., Xiang, E. W., Liu, N. N., and Yang, Q. (2010). *Transfer Learning in Collaborative Filtering for Sparsity Reduction*. In Fox, M. and Poole, D. (eds.): *AAAI*: AAAI Press. (Cited on page 35.)

[PARK and FADER 2004] Park, Y. H. and Fader, P. S. (2004). *Modeling browsing behavior at multiple websites*. Marketing Science, pp. 280–303. (Cited on pages 46 and 92.)

[PAZZANI 1999] Pazzani, M. J. (1999). *A Framework for Collaborative, Content-Based and Demographic Filtering*. Artif. Intell. Rev., 13(5-6), 393–408. (Cited on page 6.)

[PEIS et al. 2008] Peis, E., del Castillo, J. M. M., and Delgado-Lopez, J. A. (2008). *Semantic Recommender Systems. Analysis of the state of the topic*. Hipertext.net, 6. (Cited on page 34.)

[PLATT 2000] Platt, J. (2000). *Probabilistic outputs for support vector machines and comparison to regularized likelihood methods*. In Smola, A. J., Bartlett, P., Schoelkopf, B., and Schuurmans, D. (eds.): *Advances in Large Margin Classifiers*, pp. 61–74: MIT Press. (Cited on page 109.)

[POON and DOMINGOS 2006] Poon, H. and Domingos, P. (2006). *Sound and efficient inference with probabilistic and deterministic dependencies*. AAAI. (Cited on page 156.)

[RAZMERITA et al. 2003] Razmerita, L., Angehrn, A., and Maedche, A. (2003). *Ontology-based User Modeling for Knowledge Management Systems*. In *Proceedings of the 9th International Conference on User Modeling*, UM'03, pp. 213–217, Berlin, Heidelberg: Springer-Verlag. (Cited on page 178.)

[RESNICK and VARIAN 1997] Resnick, P. and Varian, H. R. (1997). *Recommender Systems*. Commun. ACM, 40(3), 56–58. (Cited on pages 3 and 33.)

[RICHARDS and MOONEY 1995] Richards, B. L. and Mooney, R. J. (1995). *Automated Refinement of First-Order Horn-Clause Domain Theories*. Mach. Learn., 19(2), 95–131. (Cited on page 159.)

[RICHARDSON and DOMINGOS 2006] Richardson, M. and Domingos, P. (2006). *Markov Logic Networks*. Machine Learning, 62, 107–136. (Cited on page 29.)

[Ruiz-Montiel and Aldana-Montes 2009] Ruiz-Montiel, M. and Aldana-Montes, J. F. (2009). *Semantically Enhanced Recommender Systems*. In *Proceedings of the Confederated International Workshops and Posters on the Move to Meaningful Internet Systems*, OTM '09, pp. 604–609, Berlin, Heidelberg: Springer-Verlag. (Cited on pages 35 and 37.)

[Salakhutdinov and Mnih 2007] Salakhutdinov, R. and Mnih, A. (2007). *Probabilistic Matrix Factorization*. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T. (eds.): *NIPS*: Curran Associates, Inc. (Cited on pages 142, 166 and 168.)

[Sarwar et al. 2000] Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. T. (2000). *Application of Dimensionality Reduction in Recommender System – A Case Study*. In *ACM Web Knowledge Discovery in Databases WEBKDD Workshop*. (Cited on pages 28, 164 and 165.)

[Schmidt-Schauss and Smolka 1991] Schmidt-Schauß, M. and Smolka, G. (1991). *Attributive Concept Descriptions with Complements*. Artificial Intelligence, 48(1), 1–26. (Cited on page 19.)

[Senkul and Salin 2012] Senkul, P. and Salin, S. (2012). *Improving Pattern Quality in Web Usage Mining by Using Semantic Information*. Knowledge and Information Systems, 30(3), 527–541. (Cited on pages 35 and 39.)

[Shi et al. 2012a] Shi, Y., Karatzoglou, A., Baltrunas, L., Larson, M., Oliver, N., and Hanjalic, A. (2012a). *CLiMF: learning to maximize reciprocal rank with collaborative less-is-more filtering*. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, pp. 139–146, New York, NY, USA: ACM. (Cited on page 124.)

[Shi et al. 2011a] Shi, Y., Larson, M., and Hanjalic, A. (2011a). *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'11, pp. 305–316, Berlin, Heidelberg: Springer-Verlag. (Cited on page 35.)

[Shi et al. 2011b] Shi, Y., Larson, M., and Hanjalic, A. (2011b). *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'11, pp. 305–316, Berlin, Heidelberg: Springer-Verlag. (Cited on pages 40, 41 and 142.)

[SHI et al. 2013] Shi, Y., Larson, M., and Hanjalic, A. (2013). *Generalized Tag-induced Cross-Domain Collaborative Filtering*. CoRR, abs/1302.4888. (Cited on pages 162, 164, 165 and 168.)

[SHI et al. 2012b] Shi, Y., Zhao, X., Wang, J., Larson, M., and Hanjalic, A. (2012b). *Adaptive Diversification of Recommendation Results via Latent Factor Portfolio*. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pp. 175–184, New York, NY, USA: ACM. (Cited on pages 134 and 136.)

[SHOVAL et al. 2008] Shoval, P., Maidel, V., and Shapira, B. (2008). *An Ontology-Content-based Filtering Method*. International Journal of Information Theories and Applications, 229, 303–318. (Cited on pages 35 and 37.)

[SIRIN et al. 2007] Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). *Pellet: A Practical OWL-DL Reasoner*. Web Semant., 5(2), 51–53. (Cited on page 22.)

[SRIVASTAVA et al. 2000] Srivastava, J., Cooley, R., Deshpande, M., and Tan, P.-N. (2000). *Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data*. SIGKDD Explor. Newsl., 1(2), 12–23. (Cited on page 4.)

[STUDER et al. 1998] Studer, R., Benjamins, V. R., and Fensel, D. (1998). *Knowledge Engineering: Principles and Methods*. Data & Knowledge Engineering, 25(1-2), 161–197. (Cited on page 14.)

[STÜHMER et al. 2009] Stühmer, R., Anicic, D., Sen, S., Ma, J., Schmidt, K.-U., and Stojanovic, N. (2009). *Lifting Events in RDF from Interactions with Annotated Web Pages*. In *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, pp. 893–908: Springer-Verlag. (Cited on page 93.)

[STUMME et al. 2002] Stumme, G., Hotho, A., and Berendt, B. (2002). *Usage Mining for and on the Semantic Web*. In *Next Generation Data Mining. Proc. NSF Workshop*, pp. 77–86, Baltimore. (Cited on page 93.)

[SZOMSZOR et al. 2008] Szomszor, M., Alani, H., Cantador, I., O'Hara, K., and Shadbolt, N. (2008). *Semantic Modelling of User Interests Based on Cross-Folksonomy Analysis*. In *Proceedings of the 7th International Conference on The Semantic Web*, ISWC '08, pp. 632–648, Berlin, Heidelberg: Springer-Verlag. (Cited on page 35.)

[TANG et al. 2012] Tang, J., Wu, S., Sun, J., and Su, H. (2012). *Cross-domain Collaboration Recommendation*. In *Proceedings of the 18th ACM SIGKDD*

International Conference on Knowledge Discovery and Data Mining, KDD '12, pp. 1285–1293, New York, NY, USA: ACM. (Cited on page 35.)

[TANG et al. 2011] Tang, T., Winoto, P., and Ye, R. (2011). *Analysis of a multi-domain recommender system*. In *Data Mining and Intelligent Information Technology Applications (ICMiA), 2011 3rd International Conference on*, pp. 280 –285. (Cited on page 35.)

[THIAGARAJAN et al. 2008] Thiagarajan, R., Manjunath, G., and Stumptner, M. (2008). *Computing semantic similarity using ontologies*: Technical Report, HP Labs. `http://www.hpl.hp.com/techreports/2008/HPL-2008-87.pdf` (accessed Apr. 10, 2014). (Cited on page 109.)

[TSOCHANTARIDIS et al. 2004] Tsochantaridis, I., Hofmann, T., Joachims, T., and Altun, Y. (2004). *Support vector machine learning for interdependent and structured output spaces*. In *Proceedings of the 21. International Conference on Machine learning*, NY, USA: ACM. (Cited on page 76.)

[TUDORACHE et al. 2013] Tudorache, T., Nyulas, C., Noy, N. F., and Musen, M. A. (2013). *WebProtégé: A Collaborative Ontology Editor and Knowledge Acquisition Tool for the Web*. Semantic Web Journal, 4(1), 89–99. (Cited on page 22.)

[TVAROŽEK et al. 2007] Tvarožek, M., Barla, M., and Bieliková, M. (2007). *Personalized Presentation in Web-Based Information Systems*. In *Proceedings of the 33rd conference on Current Trends in Theory and Practice of Computer Science*, pp. 796–807, Berlin, Heidelberg: Springer-Verlag. (Cited on page 93.)

[VANZIN et al. 2005] Vanzin, M., Becker, K., and Ruiz, D. D. A. (2005). *Ontology-Based Filtering Mechanisms for Web Usage Patterns Retrieval.*. In *EC-Web'05*, pp. 267–277. (Cited on page 94.)

[VAPNIK 1995] Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer New York Inc. (Cited on page 26.)

[VARGAS and CASTELLS 2011a] Vargas, S. and Castells, P. (2011a). *Rank and relevance in novelty and diversity metrics for recommender systems*. In *Proceedings of the 5th ACM Conference on Recommender systems*, RecSys '11, pp. 109–116, New York, NY, USA: ACM. (Cited on pages 112 and 114.)

[VARGAS and CASTELLS 2011b] Vargas, S. and Castells, P. (2011b). *Rank and Relevance in Novelty and Diversity Metrics for Recommender Systems*. In *Proceedings of the 5th ACM Conference on Recommender Systems*, RecSys '11, pp. 109–116, New York, NY, USA: ACM. (Cited on pages 134 and 135.)

[WANG and KONG 2007] Wang, R. and Kong, F. (2007). *Semantic-Enhanced Personalized Recommender System*. In *Proceedings of the Int. Conference on Machine Learning and Cybernetics*, pp. 4069–4074. (Cited on pages 35 and 39.)

[WANG et al. 2012] Wang, W., Chen, Z., Liu, J., Qi, Q., and Zhao, Z. (2012). *User-based Collaborative Filtering on Cross Domain by Tag Transfer Learning*. In *Proceedings of the 1st International Workshop on Cross Domain Knowledge Discovery in Web and Social Network Mining*, CDKD '12, pp. 10–17, New York, NY, USA: ACM. (Cited on page 35.)

[WANG 2011] Wang, Y. (2011). *Semantically-Enhanced Recommendations in Cultural Heritage*. In *Proceedings of the 2008 ACM conference on Recommender systems*: Technische Universiteit Eindhoven. (Cited on pages 35 and 37.)

[WANG et al. 2009] Wang, Y., Stash, N., Aroyo, L., Hollink, L., and Schreiber, G. (2009). *Semantic relations for content-based recommendations*. In *Proceedings of the 5th International Conference on Knowledge Capture*, pp. 209–210. (Cited on pages 35 and 37.)

[WINOTO and TANG 2008] 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*. New Generation Computing, 26(3), 209–225. (Cited on pages 34, 35, 40, 103 and 143.)

[XU et al. 2010] Xu, Z., Tresp, V., Rettinger, A., and Kersting, K. (2010). *Social network mining with nonparametric relational models*. In *Proceedings of the Second international conference on Advances in social network mining and analysis*, SNAKDD'08, pp. 77–96, Berlin, Heidelberg: Springer-Verlag. (Cited on page 142.)

[YILMAZ and SENKUL 2010] Yilmaz, H. and Senkul, P. (2010). *Using Ontology and Sequence Information for Extracting Behavior Patterns from Web Navigation Logs*. In *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, pp. 549 –556. (Cited on page 93.)

[ZHANG and HURLEY 2008] Zhang, M. and Hurley, N. (2008). *Avoiding Monotony: Improving the Diversity of Recommendation Lists*. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, RecSys '08, pp. 123–130, New York, NY, USA: ACM. (Cited on page 134.)

[ZHANG and KOREN 2007] Zhang, Y. and Koren, J. (2007). *Efficient bayesian hierarchical user modeling for recommendation system*. In *Proceedings of the*

30th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR ’07, pp. 47–54, New York, NY, USA: ACM. (Cited on page 141.)

[ZHANG et al. 2010] Zhang, Y., Cao, B., and Yeung, D.-Y. (2010). *Multi-Domain Collaborative Filtering*. In Grünwald, P. and Spirtes, P. (eds.): *UAI*, pp. 725–732: AUAI Press. (Cited on pages 35 and 142.)

[ZHAO et al. 2013] Zhao, L., Pan, S. J., Xiang, E. W., Zhong, E., Lu, Z., and Yang, Q. (2013). *Active Transfer Learning for Cross-System Recommendation*. In desJardins, M. and Littman, M. L. (eds.): *AAAI*: AAAI Press. (Cited on page 35.)

[ZHOU et al. 2004] Zhou, B., Hui, S., and Chang, K. (2004). *An Intelligent Recommender System Using Sequential Web Access Patterns*. In *IEEE Conference on Cybernetics and Intelligent Systems*, pp. 393–398. (Cited on pages 35 and 36.)

[ZIEGLER et al. 2004] Ziegler, C.-N., Lausen, G., and Schmidt-Thieme, L. (2004). *Taxonomy-driven computation of product recommendations*. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, CIKM ’04, pp. 406–415, New York, NY, USA: ACM. (Cited on pages 35 and 38.)

[ZIEGLER et al. 2005] 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 ’05, pp. 22–32, New York, NY, USA: ACM. (Cited on pages 113, 134 and 162.)

List of Figures

List of Tables

List of Algorithms