

DM

Implementing Profile Dynamics

MASTER DISSERTATION

Katherin Fátima de Ponte Varela

MASTER IN INFORMATICS ENGINEERING



UNIVERSIDADE da MADEIRA

A Nossa Universidade

www.uma.pt

September | 2023

Implementing Profile Dynamics

MASTER DISSERTATION

Katherin Fátima de Ponte Varela

MASTER IN INFORMATICS ENGINEERING

ORIENTATION

Eduardo Leopoldo Fermé

CO-ORIENTATION

Diogo Nuno Teixeira Freitas



FACULTY OF EXACT SCIENCES AND
ENGINEERING

MASTER'S IN COMPUTER ENGINEERING

Implementing Profile Dynamics

Katherin Varela

Guided by:
Eduardo Fermé
Diogo Nuno Freitas

December 20, 2023

To my family.

Implementing Profile Dynamics

Abstract

As technology continues to evolve, the abundance of information available to users has significantly expanded. This growth has paralleled an increasing need to tailor user interactions and sift through irrelevant data. User Profiles, typically defined as compilations of pertinent attributes about users and their interests, are integral to the process of personalization.

This dissertation is a component of the broader KBAI Smart Ed (KBAI-SE) project and centers on the creation of a dynamic user profiling algorithm. This algorithm is designed to operate within three distinct use cases: project management, news recommendation, and tourism activity recommendation. Additionally, the dissertation will encompass an article that comprehensively covers the current state of the art in constructing and maintaining these profiles, referred to as User Modeling. This paper will have a primary focus on the elements requisite for establishing a dynamic process capable of adapting to the evolution of users over time, a phenomenon termed Profile Dynamics. Moreover, it will explore the diverse categories of systems in which User Profiles can be implemented to enable effective personalization.

Keywords: user profiles, user profile modeling, personalization, profile dynamics

Implementing Profile Dynamics

Resumo

À medida que a tecnologia continua a evoluir, a abundância de informação disponível para os utilizadores aumentou significativamente. Este crescimento foi acompanhado por uma necessidade crescente de personalizar as interações dos utilizadores e de filtrar os dados irrelevantes. Os perfis de utilizador, tipicamente definidos como compilações de atributos pertinentes sobre os utilizadores e os seus interesses, são parte integrante do processo de personalização.

Esta dissertação é uma componente do projeto mais vasto KBAI-SE e centra-se na criação de um algoritmo dinâmico de criação de perfis de utilizador. Este algoritmo foi concebido para operar em três casos de utilização distintos: gestão de projectos, recomendação de notícias e recomendação de actividades turísticas. Adicionalmente, a dissertação englobará um artigo que cobre de forma abrangente o estado atual da arte na construção e manutenção destes perfis, designado por Modelação de Utilizadores. Este artigo terá como foco principal os elementos necessários para estabelecer um processo dinâmico capaz de se adaptar à evolução dos utilizadores ao longo do tempo, um fenómeno denominado Dinâmica do Perfil. Além disso, o artigo explorará as diversas categorias de sistemas em que os perfis de utilizador podem ser implementados para permitir uma personalização eficaz.

Palavras-chave: perfis de utilizador, modelação de perfis de utilizador, personalização, dinâmicas de perfis

Acknowledgements

First and foremost, I would like to extend my deepest appreciation to my advisors, Professor Eduardo Fermé and Professor Diogo Freitas. They have consistently been available to provide guidance, review the contents of this dissertation, and offer valuable suggestions.

I am profoundly grateful to my family, specially my fiancé, whose unwavering support has been instrumental in the development of this work. Their encouragement and understanding have been invaluable throughout this journey, and I dedicate this dissertation to them.

I also want to express my gratitude to my colleagues, Diogo Gouveia and Rodrigo Vieira, and the KBAI Research team for their collaboration and support.

Furthermore, I would like to acknowledge all my professors at the Faculty of Exact Sciences and Engineering. Their dedicated teaching and mentorship have equipped me with the knowledge and methodologies that I possess today.

This work was partially supported by project M1420-01-0247-FEDER-000073 in the Operational Program for the Autonomous Region of Madeira (Madeira 14—20).

Contents

1	Introduction	1
1.1	Problem description	1
1.2	The KBAI-SE project	3
1.3	Proposal	4
1.3.1	Information Retrieval Phase	5
1.3.2	Profile Construction Phase	5
1.3.3	Profile Modeling and Selection Phase	5
1.3.4	Dynamization Phase	6
1.4	Dissertation structure	6
2	State-of-the-Art	7
2.1	Introduction	7
2.2	User Profiles Modeling	8
2.2.1	Step 1—Information Retrieval	10
2.2.2	Step 2—Feature Representation	12
2.2.3	Step 3—User Profile Construction	15
2.2.4	Step 4—Preference Learning	15
2.2.5	Step 5—Profile Dynamics	17
2.3	Applications	20
2.3.1	Recommender Systems	21
2.3.2	Personalized/Adaptive Systems	26
2.4	Conclusions	27
3	Implementation	28
3.1	Tools used	28
3.2	Development phase	29
3.3	General structure	33
3.4	Use cases	33
3.4.1	Tourism	33
3.4.2	News	34
3.4.3	Project Management	34
3.5	System structure	34
3.5.1	Explicit Collection Module	36

CONTENTS

3.5.2	Implicit Collection Module	37
3.5.3	Recommendation Module	38
3.5.4	Dynamization Module	39
4	Conclusions and Future Work	42
4.1	Main contributions	43
4.2	Future work	43
Appendices		
A	Implicit Collection Module	54
A.1	Propagation module	54
A.2	Demographic clustering module	55
B	Recommendation module	59
C	Dynamization module	63
D	Article database	66
D.1	Excel table	66
D.2	Litmaps articles	81

List of Figures

1.1	General project structure.	2
1.2	User profile framework.	4
2.1	Distribution of the papers analyzed by topic.	9
2.2	User modeling phases required to build user profiles [RL08].	10
2.3	Highlighted papers of Information Retrieval.	12
2.4	Highlighted papers of Feature Representation.	15
2.5	Highlighted papers of Profile Dynamics.	21
2.6	Word cloud showing the most frequently used terms in the extracted papers.	27
3.1	Example of an interest profile	30
3.2	Example of an item profile	31
3.3	Recommendation process example	32
3.4	Over-ranked items example	32
3.5	System structure	35
3.6	Explicit collection platform	36
3.7	Dynamization module example	41
D.1	All articles	81
D.2	Adaptive systems	82
D.3	Profile Dynamics	82
D.4	Recommender Systems	83
D.5	User Profile Modeling	83
D.6	Personalized Systems	84

Listings

A.1	Propagation function for interests	54
A.2	Modify value of implicit weight of target interest	54
A.3	Calculate mean between two values	54
A.4	Modify value of implicit weight of parent and child interests of target interest	54
A.5	Modify implicit weights of affinity interests	55
A.6	Update explicit and implicit weight of the item	55
A.7	Create demographic clusters and infer preference weights	55
A.8	Encode labels	56
A.9	Normalize values	56
A.10	Use k-means to create and assign clusters	56
A.11	Calculate average preference level for each category by cluster	57
A.12	Add average weights	57
A.13	Infer missing preference weights of customers based on the average cluster weights	58
B.1	Recommender module	59
B.2	Change values based on session context	59
B.3	Modify characteristic values based on session context	59
B.4	Modify interest values based on session context	60
B.5	Modify item values based on session context	60
B.6	Filtering module	60
B.7	Filter items by characteristic	60
B.8	Filter items by vetted interest	61
B.9	Filter items by vetted item	61
B.10	Measure distance between filtered items and user profile	61
B.11	Return top recommended items	61
B.12	Add recommended items to session	61
C.1	Analyze latest user feedback to modify interest weights	63
C.2	Analyze recommended items in each session	63
C.3	Save interest categories that will be modifies	64
C.4	Count instances of interest categories	64
C.5	Dynamization algorithm	65

List of Tables

2.1	A list of the most used Collaborative Filtering (CF) techniques [BOHG13, AT05].	22
-----	---	----

Acronyms

AI Artificial Intelligence. 3

API Application Programming Interface. 2

CB Content-Based. 21, 23, 24

CE Collaborative Evolution. 23

CF Collaborative Filtering. ix, 21–24

CNN Convolutional Neural Network. 14

CS1 Case study 1. 1

CS2 Case study 2. 1

CS3 Case study 3. 2

DB Database. 5

IPF Injected Preference Fusion. 14

KB Knowledge-Based. 21, 24

kb Knowledge Base. 1–3, 6, 28, 29, 32, 33, 42

KBAI Knowledge-Based Artificial Intelligence. 3

KBAI-SE KBAI Smart Ed. ii, iii, v, 1–3, 6, 29, 30, 33, 42

MF Matrix Factorization. 14, 23

NLP Natural Language Processing. 14

ODP Open Directory Project. 13

POI Point of Interest. 13

Acronyms

RS Recommender Systems. 7, 8, 21, 25, 27

TF-IDF Term Frequency–Inverse Document Frequency. 13

Chapter 1

Introduction

This dissertation is integrated into the KBAI-SE project, which aims to create an interactive tool for engaging with a dynamic and collaborative Knowledge Base (kb), as well as for the rational processing of accumulated knowledge. Supported by a central platform, KBAI-SE facilitates the registration and correlation of various concepts.

The objective of this dissertation is to develop a comprehensive framework for dynamically modeling user profiles and employing them across diverse real-world applications. To realize this objective, the project is structured into five distinct phases.

The first phase is the information collection stage, during which methodologies for gathering user data will be defined. The second phase involves profile construction, where strategies for representing user profiles will be outlined. The third phase, profile modeling and selection, entails developing algorithms to identify and assess users' interests, along with mechanisms for selecting items with the highest scores. Lastly, the fourth phase is the dynamization stage, focusing on devising rules to capture user changes and evolutionary patterns.

1.1 Problem description

This dissertation focuses on the creation of a dynamic user profiling algorithm. This algorithm is designed to model a user's preferences through their interactions with the system and the information they provide. Additionally, it is capable of evolving over time by detecting shifts in a user's interests and preferences. To achieve this, a series of rules grounded in Belief Revision principles will be formulated, ensuring the algorithm's objectives are effectively met. The developed algorithm will subsequently be applied to three distinct real-world use cases:

1. Case study 1 (CS1)—project management, where all the knowledge about a project (e.g., workers and effort estimation) will be saved and managed to accomplish its tasks efficiently.
2. Case study 2 (CS2)—news recommendation, where users will get news recommended to them according to their interests in different news categories and events.

1.1. PROBLEM DESCRIPTION

3. Case study 3 (CS3)—tourism activities recommendation, where users will be recommended tourism activities according to, e.g., their location and interests.

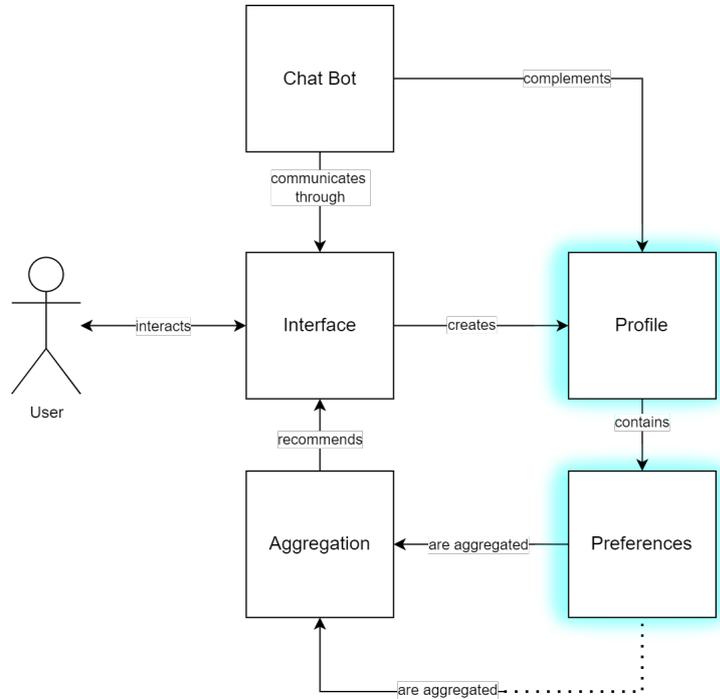


Figure 1.1: General project structure.

This dissertation is inserted in the KBAI-SE project, fig. 1.1 shows the structure of the project, with the highlighted parts corresponding to the work that will be developed in this document. The user will interact with the interface, specifying what they want from the system by communicating through a chatbot, these interactions will be used to create the user profile through the information given manually by the user through the system’s graphic interface or by inferring the user’s preferences based on defined key interactions, all of this information will be sent to the user profiling algorithm through the kb Application Programming Interface (API). These preferences will then be aggregated in the aggregation module to calculate group preference to be able to make group recommendations.

One of the primary requirements for this project is the in-depth understanding of the user’s needs and interests to provide accurate item recommendations based on their platform interactions.

It is important to acknowledge that user preferences are not static and evolve over time, making it essential to accurately identify these changes for precise recommendations. However, this can be challenging, especially when certain interests are constrained to short periods [HF15] or specific contexts [WHW⁺22], requiring the ability to perceive when these changes occur.

Additionally, collecting information about user interests poses difficulties. This information can be collected explicitly or implicitly. Explicit information, entered directly by the

user, is the most useful and accurate but can be time-consuming and dependent on the user's understanding of the preference concepts and rating scale [DSRO20]. On the other hand, implicit information, gathered passively through user interactions or by inferring preferences from similar users, is relatively easier to obtain but may not be as accurate as explicit data due to the inference process [MIM10].

The main objective of this dissertation is to study existing methods for creating and maintaining user profiles for personalized systems. These methods will be carefully analyzed and classified to identify the most effective approach for calculating profiles based on user interactions and knowledge-based models to meet the previously mentioned requirements.

Subsequently, these techniques will be implemented in various case studies within the project's scope, including tourism, project management, and news domains. This implementation will involve combining explicitly collected information, user interaction history, and knowledge extrapolation from similar customers. The ultimate goal is to predict the potential interests and motivations of new customers with varying degrees of confidence.

1.2 The KBAI-SE project

The KBAI Smart Ed project aims to develop an intelligent knowledge management application comprising a knowledge editing product enriched with an automatic reasoning engine, KBAI stands for Knowledge-Based Artificial Intelligence.

The KBAI-SE project will conduct research and experimental development to integrate academic and scientific knowledge from research centers with the expertise and experience of professionals in the team. The KBAI-SE product will serve as an interactive tool for a dynamic and collaborative kb, capable of rational processing of accumulated knowledge.

The central component of the system is envisioned to integrate various reasoning mechanisms to facilitate automatic knowledge updating (e.g., verifying its accuracy) and generating new knowledge through learning or logical inference.

The primary focus of the project lies in exploring Artificial Intelligence (AI) theories and tools, their current level of maturity, and their potential applications for the intended purposes. The research will revolve around two main axes: (a) the study of Reasoning Algorithms and Knowledge Representation techniques that are already at a stage of maturity suitable for simple and efficient use, and (b) finding intuitive visual methods for consulting and editing knowledge to streamline the process for users to input their knowledge into the system and execute queries effectively.

While these theories are well-developed academically, their effective application in business or personal contexts necessitates validation to ensure compatibility with the data structures chosen for KBAI-SE and to assess if processing requirements are efficient enough for timely user interactions or background processing.

To achieve this, a comprehensive survey of the state of the art will be conducted to gauge maturity, applicability to the defined form of knowledge representation, and the availability of libraries suitable for business contexts. Within the scope of industrial research, an analysis of the current data structure and the requirements introduced by each reasoning theory will

be carried out, followed by a comparison with data repositories available for their implementation (e.g., graph-oriented databases). Based on this analysis, complementary modules will be implemented to incorporate each reasoning theory into the final product.

1.3 Proposal

The dynamic profiles will be created using a similar framework to the one developed in [HF15] and represented in Figure 1.2. The information collection process about the user will be done explicitly and implicitly. Then, the level of preference the user has for each possible option will be represented through a copy of the ontology of the possible items that will be stored in each profile. This level of preference will be calculated, and the items will be selected through a modified Hamming distance. Finally, the profile will become dynamic through the use of formal languages and a belief change-inspired approach similarly to [FGD⁺23].

The project will be divided into four phases: Information retrieval, profile construction, profile modeling, and selection and dynamization. The following subsections will describe, in-depth, each of these phases.

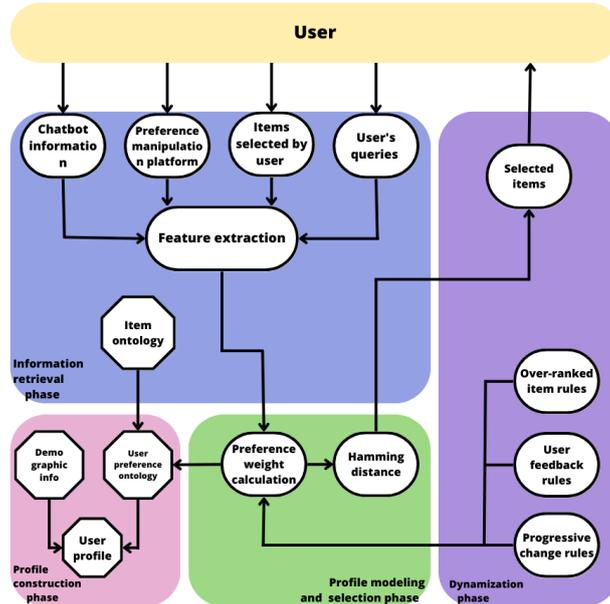


Figure 1.2: User profile framework.

1.3.1 Information Retrieval Phase

In this phase, all the implicit and explicit information about the user will be collected, and the features will be extracted. The information will be collected explicitly through the following:

- An initial form is to be answered by the user through a chat bot. This form will collect demographic data and information about users' interests.
- A platform inside the system for the users to manipulate their preferences. Here, its structure will be defined as well as the way the users' manipulation of the preferences in the platform will manipulate the weights of the preferences inside the system.

On the other hand, the implicit information retrieval will happen through:

- Users' interactions with the chat bot. Also, relevant interactions will be defined (e.g., when a user searches for a specific item), as well as the way to save these interactions in the Database (DB).
- History of all items selected by the user.
- Users' query history.

1.3.2 Profile Construction Phase

In this phase, it will be defined the way the profile is going to be represented. The first step will be the definition of the structure of the ontology of the items in the system, defining the relationships between the concepts. It is important to note here that this structure will be generic and will have to work for all the use cases. Finally, the ontology will be developed according to the structure defined.

After the creation of the item ontology, the structure of the user profile will be created. For that, a copy of the item ontology will be created to add to the user profile structure. This copy will then be used so each concept in the copy ontology will be paired with the level of preference the user has for that concept (i.e., it will form the preferences profile of the user). Then the demographic data of the user will be added, and the final user profile structure will be defined.

1.3.3 Profile Modeling and Selection Phase

In this phase, the weights of users' preferences will be calculated, and the items with a higher level of preference will be selected. Firstly, the techniques that will be used to calculate the preferences will be developed. Additionally, the rules that will manipulate the weights and the algorithms that will calculate them will be defined.

Afterward, the pre-filtering of options according to context will be defined. For that, the context values and their relationships with the features that will be affected by them will be specified. The rules that establish where the context affects the browsing process will be stipulated, and the filtering will be implemented according to the context.

Finally, the items with higher preference weights will be selected using a modified Hamming distance. Tests will be then developed to confirm the efficiency of the selections and the preference values.

1.3.4 Dynamization Phase

This phase will consist of the development of Belief Change rules that will define how the profile will change over time. First of all, the rules that will modify the preference weights based on over-ranked items will be defined; secondarily, the rules that will permute the user profile based on users' feedback will be developed. Care will be taken to make the profile's dynamization a gradual process; that is, avoiding rules that cause abrupt changes.

This phase will be finalized with the development and performance of tests to determine the effectiveness of the rules for the dynamization of the profile.

Overall, Phase 1 will collect users' demographic data (e.g., gender, age, education, nationality). Phase 2 will, on the other hand, create the general structure for all ontologies, while Phase 3 will develop the general algorithms to calculate the level of preference. Lastly, Phase 4 will be generic for all use cases through the development of general dynamic rules.

1.4 Dissertation structure

The dissertation is structured as follows. After the introduction in Chapter 1, Chapter 2 reviews the state-of-the-art, displaying the latest user profiling techniques and highlighting the elements to take into account when modeling user preferences such as the dynamics of user interests or the explainability of recommendations. From this chapter, an article was extracted and sent to the Artificial Intelligence Review Journal.

Chapter 3 develops an extensive depiction of the dynamic user profiling algorithm and its subsequent implementation. This chapter introduces the tools employed for the algorithm's development and provides rationale behind their selection. It delineates a breakdown of the kb's structure, expounding upon its constituents and their interrelationships. The chapter delves into the algorithm's operational mechanism, elucidating how it utilizes users' demographic information and interactions within the KBAI-SE platform to deduce user preferences. Furthermore, it elucidates the algorithm's capability to adapt preference values over time, in accordance with the evolving interests manifested by users through their recent interactions with the system. Ultimately, the chapter concludes by outlining the various use cases in which the algorithm is to be applied.

Chapter 4 will outline the conclusions and future work for this project.

Chapter 2

State-of-the-Art

2.1 Introduction

User profiles consist of a collection of characteristics that describe how users interact with a given system, encompassing their interests and needs [AS99]. These characteristics may include demographic traits, preferences, and desires, among others. User profiles are established and managed through a process known as User Profile Modeling, which involves two fundamental steps. The initial step entails defining what information about the user should be stored and how it should be represented. Subsequently, the second step involves determining the methodology for constructing, updating, and maintaining users' profiles [LMNS19, p. 9].

User modeling is essential for personalized systems, especially given the current data explosion where the amount of digital data being generated and stored is growing exponentially. Through user modeling, personalized systems acquire insights into users' characteristics and can adapt their behavior dynamically. This adaptation includes tailoring content to individual users or user groups based on their behavior, needs, and preferences. It's worth highlighting that user modeling can also serve the purpose of uncovering and introducing users to new interests that may align with their preferences [Mou97].

More specifically, one of the primary applications of user modeling is information filtering, which finds use in Recommender Systems (RS) across various domains. These domains encompass web personalization, computational advertising, e-learning systems [LNNH09, HF15, YCC⁺15, WHW⁺22], and search engines [ENSN19]. RS refers to systems that gather information about user preferences for specific items (such as movies, songs, and books) and subsequently select new items based on their characteristics. These selected items are chosen with the aim of aligning with the user's interests and needs [BOHG13, IFO15].

Prediction systems can also reap the advantages of user modeling, particularly in estimating a user's rating for a specific item when explicit user feedback is absent [FEMR18]. Furthermore, in adaptive systems, user modeling is employed to customize the user interface. This includes adjustments like changing font size or displaying context-aware information, all aimed at enhancing the effectiveness of communication between the system and the user [Bro16].

Understanding the applications of user modeling highlights, however, the fact that user

profiles may change over time. In order to address this, techniques of Profile Dynamics were developed to account for the dynamic nature of users' interests and needs. In fact, the presence of only static information within user profiles (i.e., static profiles) introduces the risk of having user profiles with outdated information, and consequently, ineffective system personalization [HF15]. Therefore, profile dynamics enable the system to detect short-term or long-term changes in user characteristics and integrate them into their profiles, providing up-to-date and relevant personalized system recommendations.

It is important to note that information about users' characteristics can be acquired either explicitly, typically by collecting users' ratings, or implicitly, commonly by monitoring users' behavior. This includes monitoring songs heard, downloaded applications, websites visited, and books read [IFO15].

The aim of this work is to provide a systematic review of the existing literature in User Profiling and its Dynamics. The process of searching and retrieving publications was conducted in March 2023. There were no restrictions on the publication date. 86 publications were selected and full texts were extracted and analyzed. It is noteworthy that, from the 86 publications, 43 were conference articles, 37 were journal articles, 3 were books, and 3 were book chapters.

A comprehensive database containing all the information related to the articles has been created and is illustrated in Appendix D. In this appendix, Appendix D.1 provides an Excel table that encompasses the data of all the articles, while Appendix D.2 presents a Litmaps collection. This Litmaps collection visually demonstrates the citation relationships between the articles, offering a clear overview of the scholarly connections within the research literature.

The publications have been grouped into five categories based on the research hypothesis of each study: User Profiles Modeling ($n = 44$), Profile Dynamics ($n = 44$), Recommender Systems ($n = 41$), Personalized Systems ($n = 22$), and Adaptive Systems ($n = 23$). This taxonomy, as depicted in Figure 2.1, helps the reader understand the diverse fields of user modeling and profile dynamics. This categorization also improves the clarity and organization of the review, making it easier to navigate and comprehend the extensive body of research in this area. In the various sections of the study, we will spotlight some of the most notable papers within the respective topic of that section.

This article is structured as follows: In Section 2.2, the definition of users' profiles and the process of profile modeling is presented; in Section 2.2.5, the techniques for profile dynamics are introduced; section Section 2.3 describes the types of systems that make use of user profiles, it is divided into two sub-sections: Section 2.3.1, defines RS and their sub-classifications; Section 2.3.2 describes personalized/adaptive systems and finally Section 2.4 presents some concluding remarks.

2.2 User Profiles Modeling

User profiles play a fundamental role in RS [IFO15] and personalized or adaptive systems [BKN07]. They provide a comprehensive representation of individual users or user groups. A user profile is an instantiation of a user model within the system [ENSN19]. Typically, user profiles

2.2. USER PROFILES MODELING

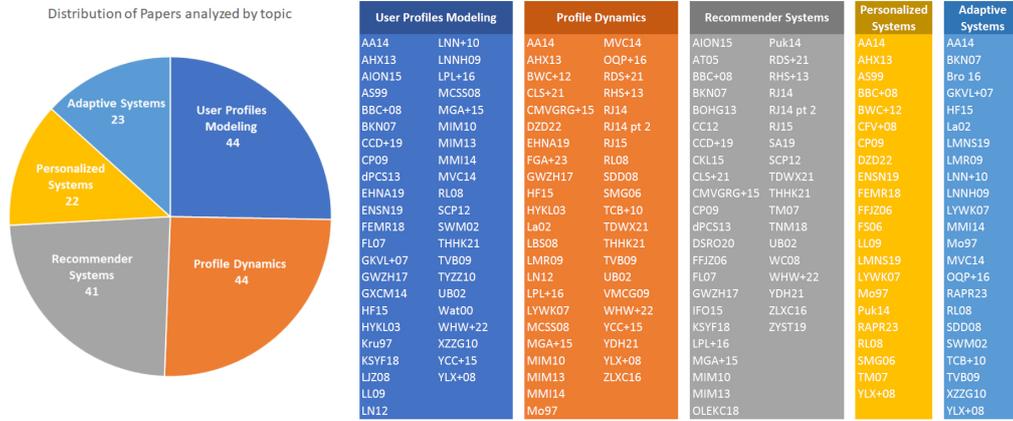


Figure 2.1: Distribution of the papers analyzed by topic.

consist of two main components: demographic information, such as age, gender, and occupation, and a set of keywords or concepts associated with corresponding values. These keywords and concepts serve to represent and estimate user intentions, interests, and other pertinent information essential for profile modeling, both for long-term and short-term time frames [DZD22]. However, some profiles can be more complex using users' behavioral data, such as clicking behavior or time spent on a web page [AHX13].

While there is a consensus in the literature that a profile represents a set of characteristics for individual users or user groups, there exists a gap when it comes to a formal definition of what constitutes a profile and its inherent structure. An exception to this gap can be found in [FGD⁺23], where a user profile $P_{\mathcal{L}}$ and a language for managing profile dynamics are rigorously defined. In this context, a profile P is defined as a tuple $\langle\langle label_1, \dots, label_n \rangle\rangle$, where $label_i \in \mathbb{N}_0$. Informally, each element within the profile tuple represents a characteristic that can assume a finite number of possible values. While natural numbers were used to define the content of each $label_i$, it's worth noting that this definition can be adapted to incorporate linguistic labels. For example, if $label_1$ represents marital status, possible values such as "single/married/separated/widowed," among others, can be utilized. To illustrate this profile structure, consider a simple example of a profile structure: $\langle\langle age, weight, height \rangle\rangle$, where a possible profile might be John = $\langle\langle 20, 80, 178 \rangle\rangle$.

As mentioned earlier, user profiles are generated through a process known as user modeling, which is subsequently employed for inferring unobservable information about users. The process of user modeling comprises several substeps, namely: information retrieval, profile construction, preference learning, and profile dynamics (or maintenance) [RL08]. Figure 2.2 summarizes this process. The subsequent sections provide a comprehensive description of each step.

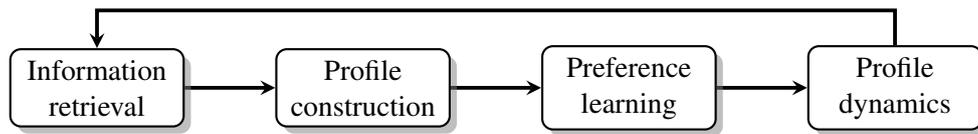


Figure 2.2: User modeling phases required to build user profiles [RL08].

2.2.1 Step 1—Information Retrieval

In order to accurately identify users’ needs and interests, the system needs to collect as much information about the user as possible; therefore, the information retrieval phase is crucial to make an effective user personalization. Information retrieval can be classified as explicit, collected by getting users’ feedback or manually editing their profile; implicit, collected by analyzing users’ interaction and behavior within the system; or hybrid, combining explicit and implicit information [IFO15].

Information about users may include demographic data, such as age, gender, and education. These data can be utilized to classify users and generate baseline recommendations based on demographically similar individuals. This technique is known as Demographic Filtering [AT05] and can be particularly useful for addressing the data sparsity problem, which occurs when limited information about users is available.

In addition to users’ demographic information, their behavior within the system, what they searched in the past, and what they liked before can be used to infer the level of interest a user has in a particular item or a specific need the user has. Nevertheless, user profiling systems usually focus solely on users’ positive interactions with the system—however, more than this information is needed to capture the whole scope of users’ interests. Knowledge about users’ negative preferences can help broaden the understanding of the user profile, allowing the system to understand not only what the user likes but also what they dislike [LL09]. Contextual information can also be used to model the user profile [RDS⁺21].

Contextual information refers to any details that can provide information about a user’s situation, such as their location, the activity they are engaged in, and their social relationships [HPV16].

The collected data undergoes a preprocessing process to extract label values, known as user profile features. These features are then employed in constructing the user profile [ENSN19], which can apply to either an individual user or a group of users.

The information used for profiling can be categorized as explicit (provided by users), implicit (automatically collected), or hybrid. In the following sub-subsections, we will delve into these techniques in detail. The highlighted papers for this subsection are presented in Figure 2.3.

Explicit Information

Explicit information retrieval relies on information explicitly provided by users. This may include demographic information, such as age, gender, and place of residence, which can be used to create demographic clusters and infer users’ preferences [Kru97].

2.2. USER PROFILES MODELING

Users can also provide positive or negative feedback information using, e.g., ratings or dislike buttons [XLW⁺20, DSRO20, DZD22]. Additionally, users can provide information about their interests [GSCM07] and manually assign weights to or order their preferences within the system.

Although simple and highly effective [ZLXC16], this information retrieval strategy has areas for improvement. The process is time-consuming and requires users to have domain knowledge of the concepts used to describe preferences and the rating scale for weighting preferences [DSRO20]. Furthermore, not all users are willing to spend time providing feedback and information to the system, even when encouraged to do so [MIM10]. For these reasons, implicit information retrieval methods have arisen.

Implicit Information

Implicit information retrieval occurs when the user information is collected automatically without the user needing to intervene in the process [BKN07]. There are many ways to implicitly collect information about users' preferences, feedback and interests.

One method is by analyzing browsing behavior, including the websites visited by the user, their contents, timestamps, and duration of the visit on each website [DZD22]. Another approach is to analyze the user's query logs, examining their past searches and creating query-flow graphs to represent the ongoing behavior of a query search. This technique is particularly useful for identifying logical sessions and recommending new queries [BBC⁺08].

Additionally, observing which options the user selects or clicks can provide valuable insights [LMNS19]. For instance, if a system presents options sorted by preference level and the user does not choose one of the top options, a comparison can be made between the selected option and the top ones to identify differences. This information can then be used to modify weights of a given model, ensuring that similar options to the selected one are prioritized in the subsequent recommendation process [ZLXC16].

This method requires, however, a considerable computational power due to the large amount of data collected, as well as data mining techniques to analyze the data. Furthermore, the implicitly collected data may be noisy [XLW⁺20], and the confidence in the obtained results is occasionally low [MIM10], especially in the initial stages. On the other hand, this information retrieval process has the advantage of being capable of collecting data continually, and thus, constantly receiving updated information [ENSN19].

Hybrid Information

Hybrid information retrieval is a methodology that leverages the strengths of both explicit and implicit feedback to verify collected information and strengthen the effectiveness of the system, particularly in the context of user profile dynamics [RJ15]. One way to obtain hybrid feedback is by using implicit feedback to verify the veracity of explicitly collected information [IFO15]. This methodology creates unbiased user models by considering both positive and negative feedback during the training phase of the model [XLW⁺20], enabling the same model to better capture the user's intention [DZD22].

2.2. USER PROFILES MODELING

Ravi et al. [RDS⁺21] argue that optimal performance of recommender systems requires the use of both implicit behavior and explicit feedback information from users. By incorporating both of these information retrieval approaches, the authors suggest that the resulting models will become more transparent and trustworthy. This increased transparency and trustworthiness is expected to encourage users to provide more information, thereby enabling the model to be better trained.

Still in the domain of recommender systems, Interactive Recommender Systems strive to integrate feedback into the recommendation process. These systems suggest items and subsequently receive feedback from users, which can be utilized to generate more refined and personalized recommendations [ZDC⁺20]. A similar application can also be found in Conversational Recommender Systems [GLH⁺21], where implicit interaction can be used as a mean to evaluate recommendation models.

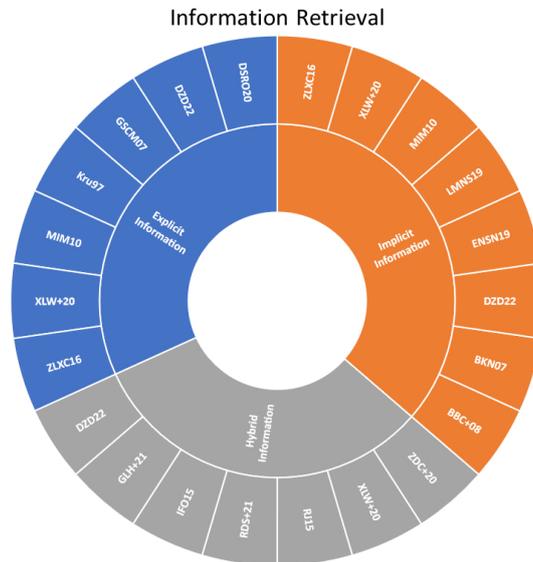


Figure 2.3: Highlighted papers of Information Retrieval.

2.2.2 Step 2—Feature Representation

After the information retrieval phase, the feature extraction phase is conducted. Feature extraction involves extracting, from the raw data, the features that can potentially be used to model the user profile. Eke et al. [ENSN19] summarized the possible features to extract depending on the application.

Following feature extraction, feature representation becomes the crucial step required for profile construction. It utilizes all the previously extracted information and constructs new user profile features that enhance profile modeling performance. Feature representation has gained increased attention, particularly since the emergence of deep learning [BHBTC22], and has been employed in various systems for different purposes.

It is important to note here that feature extraction and, later, feature representation can be applied to both users and items.

The highlighted papers of this subsection are displayed in Figure 2.4.

Text Representation

For text representation, one widely used method is the Term Frequency–Inverse Document Frequency (TF-IDF) classifier, which calculates the weight of a keyword in a document based on its frequency within that document [AT05]. For instance, in a study by Lauschke and Ntoutsis [LN12], the TF-IDF classifier was utilized to extract features in a system designed to monitor user profiles and their evolution on the social platform Twitter.

Another approach, known as the Bag-of-Words method, is a simpler alternative to TF-IDF. It analyzes the frequency of words inputted by users to generate a set of keywords representing their interests. This autoencoder technique is particularly effective in systems that rely on explicit data, such as micro-blog text [AHX13]. n -gram models [SA19] or attention models [WHW⁺22] are also commonly employed for feature extraction.

An ontology, on the other hand, is a formal knowledge representation of a specific domain in a structured manner that can be understood by both humans and machines [OLEKC18]. One of the key characteristics of ontologies is their capability to establish and describe relationships between classes [FL07]. As a result, ontologies can be considered a form of feature representation, more specifically, a representation of semantic features [RL08], which can be shared and reused in different systems. Moreover, ontologies can solve the cold-start problem, i.e., when there is not enough information about the user to make accurate user profiles [TNY17].

Ontologies represent entities, such as users, products, or services, within the system [TNM18] and concepts. Subsequently, these ontologies are compared to predict users' needs, particularly in the context of recommendation systems [WC08, RHS⁺13]. For instance, Keng et al. [KSYF18] developed a framework where the ontology structure incorporates both the Point of Interests (POIs) and the users. Recommendations are then generated by evaluating the similarity between the POIs and the users' ontologies using the Jaccard index.

Similarly, Obeid et al. [OLEKC18] implemented a semantic recommender system, wherein ontologies were employed to represent higher education institutions, employment, and students. However, machine learning techniques were utilized to generate tailored recommendations for university degrees for students.

Reference ontologies, such as Open Directory Project (ODP), also serve an important role in representing semantic features since they provide an improved and standardized framework for defining, representing, and relating entities and concepts [TNM18]. As a result, they provide a richer representation of information when compared to ontologies developed from scratch.

For example, Hawalah and Fasli [HF15] mapped web pages to concepts present in a reference ontology for web personalization. Interestingly, these authors also proposed techniques that utilize the established mapping for constructing and maintaining ontological user profiles (including session-based, long-term, and short-term profiles). These profiles were

subsequently used to adapt the system according to user behavior.

Amini et al. [AION15], on the other hand, integrated various reference ontologies into a unified and more comprehensive ontology for profiling scholar's knowledge.

Media Representation

Features from other sources such as images, text, audio, and video are often extracted and projected into latent spaces using Convolutional Neural Networks (CNNs) [BHBTC22] for user modeling. These networks are well-suited to process and model unstructured multimedia data through convolution and pooling operations [ZYST19].

For instance, in the study conducted by Hassen et al. [BHBTC22], a transfer learning technique and an autoencoder were employed to extract the latent features of the images representing items. This work was not the first work that used pretrained models to represent features at a high-level abstraction [LZE15].

Attention mechanisms are extensively employed in the field of Natural Language Processing (NLP), as well as in the extraction of visual features. These mechanisms are applied to the input data to allow the predictions models to focus on the most relevant components of the inputs [ZYST19], thereby improving the quality of predictions generated.

In that perspective, a potential strategy could involve employing two attention mechanisms to extract pertinent features associated with users and multimedia items (see, for instance, [LCLH19] and [CZH⁺17]). Subsequently, this extracted information can be incorporated into, e.g., recommendation models.

Fusion

Equally important is deep fusion methods, which have been suggested as a viable approach for modeling multiple data sources in user profiling, as shown by Farnadi et al. [FTDCM18]. This technique is capable of integrating data from multiple sources (i.e., heterogeneous information [ZYL⁺17]), such as the integration of implicit and explicit feedback [XLW⁺20], into a unified representation. This integration can occur either at the decision level or at the feature level [FPW⁺19].

Wen and collaborators [WSLM18] proposed an effective visual background recommendation for dance performances which combined textual information with visual content of images. In addition, Xiang et al. [XYZ⁺10] suggested a method named Injected Preference Fusion (IPF) for combining long- and short-term user profiles in recommendation calculations.

Still in the latent space context, Matrix Factorization (MF), or more generally, Tensor Factorization, can be used to map the user-item interactions [YDH21] in a joint latent factor space. The resulting latent factors can then be used as features to understand and predict user preferences for new items [ZYST19]. MF, gained wide popularity due to the Netflix Prize Contest [BL07], can be used for any type of feature representation [KBV09].

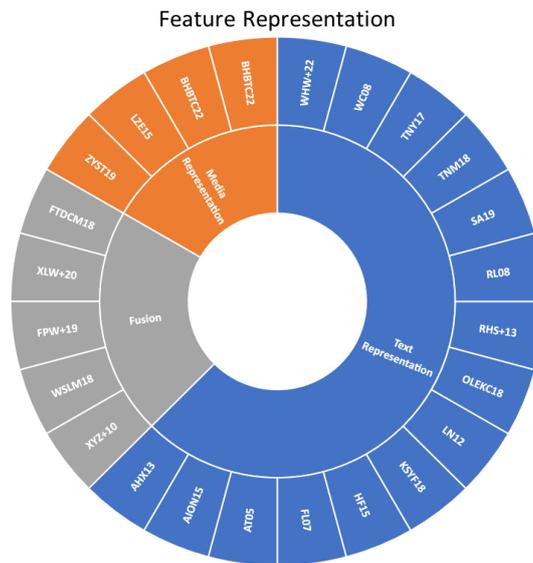


Figure 2.4: Highlighted papers of Feature Representation.

2.2.3 Step 3—User Profile Construction

In the literature, three main methods are commonly used for representing user profiles [ENSN19]. The first approach involves the creation of keyword profiles, where user profiles are constructed as vectors consisting of pairs of concepts and their respective weights [AHX13].

Another approach, known as semantic network profiles, represents user profiles as weighted semantic networks. In this method, each node within the network corresponds to a concept. This approach addresses a significant challenge found in keyword profiles, known as polysemy, which refers to situations where a single word can have multiple meanings [GSCM07].

Finally, concept profiles serve as a third approach to the representation of user profiles. Analogous to semantic network profiles, concept profiles employ weighted nodes and connections to depict concepts. Nonetheless, concept profiles differentiate themselves by employing these nodes to convey abstract subjects of interest to the user, as opposed to individual words or groups of words [GSCM07].

2.2.4 Step 4—Preference Learning

After collecting the user’s information and determining how the profile will be constructed, weights can be calculated and associated with the user’s preferences. The values of these weights can be static or dynamic. In a later part of this article, the process of profile dynamics will be explored in more detail. For now, the most common [TYZZ10] preference-learning techniques are listed below:

- Machine Learning Techniques:
 - Supervised:

2.2. USER PROFILES MODELING

- * *k*-Nearest Neighbors (KNN)—can be used to categorize users according to their interests. This method assesses users by measuring their similarity to the *k* profiles of the closest individuals, commonly referred to as the nearest neighbors. By analyzing proximity, KNN enables accurate and efficient classification of users based on their shared interests [BHBTC22].
 - * Naive Bayes (NB)—this method employs the Bayes’ theorem, a mathematical principle used to compute the probability of an event happening, given prior information about the conditions associated with the event. In the context of user profiling, NB can be applied to classify unrated Web pages [AT05], predict the web pages that are likely to capture a user’s interest [IFO15], and model user preference [BKN07, p. 390].
 - * Support Vector Machine (SVM)—powerful models work by finding the optimal hyperplane that separates data points of different classes while maximizing the margin between them. In the literature, SVMs have been employed to, e.g., update user profiles using positive and negative feedback [ZLXC16], as well as to identify relevant documents for users from the Web [TYZZ10].
- Unsupervised:
- * Agent-based—in a multi-agent system designed for user profiling, various agents collaborate and interact during different personalization phases, such as information collection [ENSN19]. Consequently, the system can identify hidden structures and capture the inherent complexity of user behavior.
 - * *k*-Means Clustering—the method is employed to cluster data points into *k* groups or clusters based on the distances between data points and their respective cluster centroids. With this in mind, the method is particularly valuable for grouping users based on their characteristics, preferences, and behaviors, thereby generating their profile [IFO15].
- Filtering Technique
 - Content-based;
 - Collaborative-based;
 - Hybrid-based;
 - Rule-based.
 - Ontology-based;
 - Neighborhood-based technique;
 - Statistical modeling.

2.2.5 Step 5—Profile Dynamics

After understanding the applications of user modeling, it is crucial to recognize that user profiles are not static; they evolve over time. To tackle this issue, Profile Dynamics techniques have been devised to accommodate the dynamic nature of users' interests and requirements. Relying solely on static information within user profiles, often referred to as static profiles, carries the risk of having outdated data in these profiles. Consequently, system personalization based on such static profiles can become ineffective [HF15]. Profile dynamics, on the other hand, empower the system to identify both short-term and long-term alterations in user characteristics and seamlessly incorporate them into their profiles. This ensures that the system can provide up-to-date and pertinent personalized recommendations, thereby enhancing user experience and system effectiveness.

The highlighted papers of this subsection are displayed in Figure 2.5.

Long-Term vs. Short-Term Interests

One way of obtaining a dynamic user profile is by creating a user profile, based on weights, for the user's long-time interests—which contains user interests that tend to be more stable over time—and another for the user's short-term interests—which represents users' current interests that change constantly. Short-term interests are usually constrained to periods of time, such as the previous month, week, or last session [HF15].

Several works have used this technique, making some relevant conclusions about the functionality of long- and short-term behavior in identifying user preferences:

- [LYWK07]– proposes a method for learning user preferences that uses independent models for short- and long-term preference. The long-term model uses taxonomic hierarchy while the short-term model uses the recent search history of the user. It uses dynamization strategies based on click-history data, considering the changes that occur in user preference.
- [BWC⁺12]– studies the interactions between long- and short-term activity, concluding that long-term interests are most helpful at the beginning of a search process. In contrast, short-term interests are more beneficial during an extended search process. Furthermore, combining both short- and long-term profiles gives better results than using either alone.
- [MIM13]– proposes a preference learning algorithm that applies dynamization through the analysis of the short-term user changes, this is, the last interaction made by the use, and long-term data, this means, the analysis of the several interactions the user had with the system, concluding that the combination of these two approaches makes the adaptation faster and more robust.
- [RDS⁺21]– presents a location recommendation framework, developing two separate agents for long- and short-term interests and giving a higher priority to the short-term agent seeing that it represents the user's current interests.

2.2. USER PROFILES MODELING

- [DZD22]– proposes a model that makes use of both positive and negative feedback to draw its conclusions, it also divides itself into long- and short-term modules with the long-term module focusing in users' long-term search logs and the short-term module focusing on the details of the search interaction of the user's current search session.

A way to enhance the weighing technique is to add the information about the context the user is inserted in at a given moment, making the system context-aware [YCC⁺15]. In other words, the system will use this information to adapt its behavior according to the user's needs in that given context and circumstance.

Temporal graph's visualization [TCJ⁺10] was also suggested to create long- and short-term profiles instead of focusing on preference weights at specific points in time.

Evolution vs. Recalculation

The evolution of the profile can be obtained in two ways [MIM10, RJ15, MMI14]:

- By recalculating it after a fixed period of time (e.g., every month). Given two profiles P_1 and P_2 and considering that between P_1 and P_2 happened R , then it can be inferred that R caused P_2 (more generally, $P_t + R = P_{t+1}$).
- By calculating the new user profile taking into account the previous profiles (evolution of profile). In this case, the following steps are taken to achieve this goal:
 1. Calculate the profiles in fixed intervals using, e.g., Machine Learning;
 2. Calculate profiles by accumulated intervals ($P_t + P_t' = P_{t+1}$);
 3. Obtain information about what happened between P_t and P_t' ;
 4. Transform that information into training rules;
 5. Repeat 3) and 4) until a stable training set is obtained.

Evolutionary/Genetic Algorithms

Evolutionary algorithms are based on the concept of evolution, such as the natural selection process, as popularized by Charles Darwin. In user profile modeling and evolution, these algorithms are used to find and update users' interests by constantly applying, e.g., genetic operators like selection, mutation, and crossover on them until an optimal solution is obtained.

El Houda et al. [EHNA19] proposed a genetic algorithm that updates the user's interests by using their queries and current interests. In the proposed algorithm, the weight of the queries or interests is used to create genes, which are then used to create chromosomes. Chromosomes are transformed by iteratively applying the genetic operators until obtaining an optimal solution (or until the stop criteria are met).

Rana and Jain [RJ14a], on the other hand, propose an evolutionary clustering algorithm called EVAR (Evolution VARiance clustering algorithm). In this algorithm, similar users are grouped in clusters that are then evolved to represent their preferences accurately over time.

Adaptation Algorithms/Rules

In this algorithm class, algorithms adapt their behavior based on the information fed to them before they are run. Several works [MIM13,ZLXC16] propose an adaptation algorithm where the options selected by the user are constantly analyzed to make items similar to the ones selected rank higher and nonselected item rank lower in the future.

Marin et al. [MIM10] introduced a similar algorithm. In their algorithm, adaptation occurs in two phases, namely: 1) an on-line phase where the preference is decreased for undesired attributes and increased in desired attributes; 2) an off-line phase where the over-ranked items collected over time are analyzed to find which characteristics appear more frequently (over a certain threshold) and then increase the preference on them.

López-Jaquero and collaborators [LJMR09] developed adaptation rules and applied them to interface development, whereas Pukkhem [Puk14] worked on a set of adaptation rules for a system in the e-learning context.

Context Awareness

Another way to achieve dynamic user profiles is through context awareness, a technique that presents the system with information about the context the user is inserted in (e.g., location). A number of studies showed a successful application of context-awareness systems:

- [WHW⁺22]– proposes a method that treats the contextual information as part of the user data, with the objective of making content-enriched models also context aware.
- [SMG06]– developed a context aware framework for automated customization of a smart home environment with multiple devices located in different areas (contexts) of the home.
- [MAC⁺07]– proposes a standard ontology that models user profiles, ontologies have been proven to be very effective in the process of modeling user context.
- [SDD08]– presents a user profile selection method that chooses a profile according to the environment the user is inserted in at the moment, it also makes use of ontologies to facilitate the modeling of the context.
- [XZZG10]– develops a method for content selection and presentation that deals with multiple types of context information, this means, not only does it deal with the most common fixed types of context information like time, device, location, etc. it also deals with other promising types like sensor data (e.g. temperature, ambient light and sound levels), or output obtained from other context-reasoning systems such as user activity recognition or mood detection.
- [RHS⁺13]– presents SMARTMUSEUM, a context aware ubiquitous recommender system for tourists that recommends cultural information, on-site, according to the tourist's needs.

2.3. APPLICATIONS

- [GXCM14]– develops a context aware personalized user modeling approach that tracks user’s digital and querying behavior to update the user preference profile.
- [YCC⁺15]– proposes a latent class statistical mixture model, named temporal context-aware mixture model (TCAM), this model uses information about users’ intrinsic interests as well as temporal context to study their social media behavior.
- [CMVGRG⁺15]– presents a context-aware, knowledge-based mobile movie recommendation system called RecomMetz. It’s context-aware approach uses the contextual information of location, time and crowd information, it’s also partially modeled through the implementation of a domain ontology primarily used in the recommendation process.

Thomsen et al. [TVB09] reiterated how systems with context awareness are a useful tool for achieving profile evolution and showed its use on medical professionals.

Belief Revision

The AGM belief revision model [AGM85, FH18] is a formal framework to represent the dynamics of the belief of a rational agent. To the best of our knowledge there are not many works that relate belief revision with user’s profiles. In particular, we can mentioned two papers: Fermé et al. [FGD⁺23] proposed a method, based on AGM, for creating and dynamizing user profiles to represent and review information about users’ interests. It also focuses on defining and enforcing the principle of consistently making minimal changes to the user profile. That means ensuring that changes to users’ interests do not happen abruptly and that no significant changes can happen; instead, changes to the knowledge base should be slow and steady [Lau02]. That article also proposed a way to create, represent and update user profiles by presenting and characterizing four operators to achieve profile dynamics through a belief revision-inspired approach. The second paper was proposed by Lau et al. [LS12]. In that paper, the authors developed a service recommendation agent based on belief revision logic to handle the non-monotonicity problem of web service recommendation. They applied belief revision-based reasoning to determine the most suitable context for the initial service request based on the beliefs stored in the user’s profile. After service request reasoning, the set of potential web services is identified and ranked. The highest-ranked services are considered to be the most desirable ones that match the user’s specific interests.

2.3 Applications

User profiles are used in two main types of systems, Recommender Systems and Personalized/Adaptive Systems

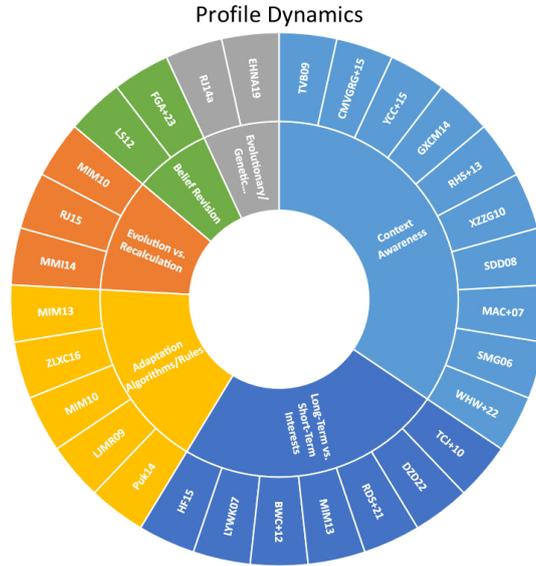


Figure 2.5: Highlighted papers of Profile Dynamics.

2.3.1 Recommender Systems

RS are systems that make use of the information provided by the user profile on the user’s interest in a particular item (e.g., movie, song, or book) to recommend other similar items to the user [BOHG13]. This study area emerged in the mid-1990s when researchers focused on predicting the ratings of items the user has yet to see.

Those systems can be classified as CF Recommendation, Content-Based (CB) Recommendation, Knowledge-Based (KB) Recommendation, and Hybrid Recommendation [AT05] according to the filtering algorithm used.

Collaborative Filtering Recommendation

CF compares the target user to all the other users and finds similarities between them. The users with greater similarity are then grouped in neighborhoods or clusters. The previous ratings of these groups are analyzed to find items that have not yet been rated by the user but might be of interest based on the positive ratings.

This is the most widely used recommendation technique and is best suited when it is desired to recommend items that are difficult to describe in an objective manner, such as movies, art pieces, or music [IFO15].

Memory-Based vs. Model-Based Methods CF can be classified as Model- and Memory-based. Memory-Based methods consist of heuristics that use the information collected about the interests stored in the user profile to make predictions (e.g., similarity metrics to obtain the distance between users and items). On the other hand, model-based methods use the interests’ information to fit a model that will later be used to make recommendations by predicting their

2.3. APPLICATIONS

effectiveness in satisfying the user’s needs [AT05,BOHG13].

Table 2.1 presents some of the most common techniques used in CF according to their classification.

Table 2.1: A list of the most used CF techniques [BOHG13,AT05].

Category	Techniques
Model-Based	Genetic Algorithms [UB02]
	Fuzzy Logic [MMI14]
	Neural Networks [WHW ⁺ 22]
	Bayesian Networks [AAO14]
	Linear Regression [CP09]
	Probabilistic Models [SA19]
	Clustering [RJ14a]
	Latent Semantic Indexing (LSI) [BOHG13]
Singular Value Decomposition (SVD) [BOHG13]	
Memory-Based	Similarity Measures [SCP12]
	Aggregation Approaches [TNM18]
	Nearest Neighbor (Cosine Correlation) [AT05]
	Clustering [SCP12]
Graph Theory [AT05]	

Benefits of Collaborative Filtering Recommendation This recommendation strategy works well when dealing with difficult items (such as nonobjective items) to describe because they depend on factors like opinions. It can also make new and effective recommendations that are not represented in the user profile by detecting interests that the target user has yet to show but similar users have already expressed.

Limitations of Collaborative Filtering Recommendation One of the main problems in CF is known as the cold-start problem, which happens at the beginning of the interaction when the system does not have enough information about the user to find their neighbors and make accurate recommendations [BOHG13].

Data sparsity is also a common problem because users do not like to spend time doing reviews and giving feedback. As a result, a lot of information is missing from the database [IFO15].

Due to the processing of vast amounts of data, scalability is also considered a problem, especially when the number of users grows. MapReduce can, however, minimize this problem [CKL15].

Finally, synonymy problems (i.e., polysemy) may occur when similar items have different names in the database but similar meanings, being challenging to distinguish similar items [IFO15].

Examples of Collaborative Filtering Recommendation The following are examples of the application of CF recommendation for real-world problems:

- [CP09]—a framework for Yahoo! News recommendation system that used CF and was able to handle the cold-start issue by adding dynamic features to the user profile.
- [RJ14a, RJ14b]—used an evolutionary clustering algorithm to identify the groups of similar users more accurately.
- [CKL15]—applied pairwise similarity of users based on an ontology model in a digital library.
- [DSRO20]—used a deep learning model for extracting product features in order to improve CF recommendation.
- [RDS⁺21]—a location recommender system based on a multi-agent framework. The authors also used long-term and short-term user behavior to make the process dynamic.
- [LPL⁺16]—proposed a method called Collaborative Evolution (CE) to integrate MF and an autoregressive vector into a unified learning framework where the user’s latent interests at every point in time are learned iteratively in a CF fashion. Then, the evolution of these interests was modeled to further guide the learning of these latent interests.

Content-Based Recommendation

This method analyzes past user interactions with the system, sees which items the user liked in the past, and searches for similar ones to recommend. This kind of recommendation is ideal for document recommendations, such as books and news.

CB systems can be classified into case-based reasoning and attribute-based technique. Case-based reasoning recommends items that are most related to items the user previously liked. On the other hand, the attribute-based technique focuses on the features of the items liked in the past and recommends new items with those same features [TNM18].

Some of the techniques used in this kind of system are Bayesian Classifiers, Clustering, Decision Trees, and Artificial Neural Networks.

Benefits of Content-Based Recommendations These systems have the benefit of not depending on the information of other users, thus relying only on the information of the target user; to adapt their recommendations quickly; to be more private because users do not have to share their profile information for the system to function; to support explainable recommendation since they base their recommendations on item features which facilitates explainability [IFO15]

Limitation of Content-Based Recommendation These systems suffer from content analysis limitations since they depend on the items being described accurately and extensively to make practical and accurate recommendations.

They can also suffer from overspecialization, which happens when users get similar items consistently and cannot discover new interests. The cold-start problem can also put into question the usability of these systems [AT05].

Examples of Content-Based Recommendation The following are examples of the application of CB recommendation for real-world problems:

- [MGA⁺15]—a recommender system that facilitates knowledge accessibility in an engineering setting that benefits from CB filtering, as well as collaborative filtering.
- [KSYF18]—a CB recommender that utilizes ontological information.

Knowledge-Based Recommendation

Makes a deep analysis of the items and uses this profound knowledge to find ways to satisfy specific user needs [FFJZ06]. This deep knowledge is sometimes accomplished by the use of ontologies to represent it because they can be used to describe objects semantically without the restraints from other models, as databases have [CMVGRG⁺15, TNM18].

Benefits of Knowledge-Based Recommendation This kind of system has an advantage over CF and CB systems. More specifically, they do not suffer from problems that stem from new users and new items (e.g., cold-start problem) or rating sparsity problems because they make use of the deep knowledge acquired on the items to calculate recommendations [AT05, TNM18].

Limitations of Knowledge-Based Recommendation Some limitations of KB systems are the need for an understanding and knowledge of engineering skills [TNM18] and the need for deep knowledge acquisition of the problem at hand. For this reason, the majority of these kinds of systems are developed for areas where domain knowledge is easily available [AT05].

Examples of Knowledge-Based Recommendation The following are examples of the application of KB recommendation for real-world problems:

- [FFJZ06]—a KB recommender called CWAdvisor developed for two use cases: A digital camera adviser and a financial service adviser.
- [CC12]—presents a study of KB systems that make use of semantic relationships between items to increase performance, describing different methods to create semantically enhanced models.
- [RHS⁺13]—a recommender for tourism sites, such as museums.
- [CMVGRG⁺15]—a KB context-aware recommender for movie show times.

Hybrid Approaches

It combines the other types of recommendation intending to overcome the problems of one of the techniques by using the strengths of another, and as a result, get more effective results [IFO15].

Explanations

Explanations on RS are techniques used to explain to the user why they were recommended something. The literature [TM07] describes seven benefits of explanations, these are:

- **Transparency**—Show the user how the system chose the recommended item.
- **Scrutability**—Give the user the ability to correct the system in the case of an ineffective recommendation (e.g., not interested button).
- **Trust**—Measure how much the user trusts the system, which can be achieved by questionnaires or by analyzing customer loyalty and sales numbers.
- **Persuasiveness**—Ability of the system to convince the user to choose an item one would typically not be interested in.
- **Effectiveness**—Help users to make the correct decision according to their preferences. Additionally, the level of satisfaction with the item before and after the purchase can be analyzed to see if it really was a good fit for the user.
- **Efficiency**—The user should be able to use the system and select recommendations quickly. This item can be measured by seeing how much the user took to complete a task
- **Satisfaction**—Aims to increase the users' satisfaction with their choices and with the system.

The following are examples of systems that make use of explanations:

- [YDH21]—a General Knowledge Enhanced Framework for Explainable Sequential Recommendation (GFE) to capture user's fine-grained preferences and evolution of dynamic preferences. Additionally, the system can offer explanations from the points of view of macrocosm and microcosm.
- [THHK21]—a User-Centric Path Reasoning (UCPR) network (UCPR), which created a framework that enables explainable recommendations by guiding the search from the aspect of the user demand, increasing explanation diversity.
- [CLS⁺21]—a Temporal Meta-path Guided Explainable Recommendation (TMER) that sequentially models dynamic user-item interaction evolution on a dynamic knowledge graph to obtain explainable recommendations.

2.3.2 Personalized/Adaptive Systems

Systems that use the information on user profiles to customize their interface and behavior according to the user's needs and preferences. It aims to provide personalized content by filtering content that the user is not interested in and showing items that the user will be more interested in first. Other changes may also include changing, e.g., the font size and style according to user preference [Bro16].

According to [FP06] personalization is composed of three dimensions:

- What is personalized— this could be the content of the information presented, how it is presented, the channel of information through which the information is delivered and the functionality of the system itself.
- Target of the personalization— this could be a determined category of users (e.g. people with pets, families with little children, etc.) or a specific individual.
- Who does the personalization— this refers to the degree in which the personalization process is automated, when the user participates actively in the process by giving information to the system to guide the adaptation it's called explicit personalization, meanwhile, if the system does the whole process automatically it's called implicit personalization.

Personalized Search

- [Mou97]—presents a system named Amalthea, which offers personalized search services through personalized information filtering.
- [AAO14]—develops a way of modelling and representing long and short-term user profiles for personalized search based on dynamic bayesian networks.
- [HF15]—proposes a personalized multi-agent search system that models its dynamic user profiles by using the web pages visited by the user to model the long and short-term user interests.

e-learning area

- [LNNH09]—proposes a generic model for providing personalized learning resources and services to students for blended learning.
- [RL08]—presents an Ontology-Based User Modelling framework that will be tested in the context of a Semantic Learning Portal where user modelling is a key aspect for these kind of systems.
- [LNN⁺10]—describes the different applications of user profiles in various adaptive systems, specifically Adaptive Instructional Systems with the objective of improving them so they can be used in a blended-learning environment.

2.4. CONCLUSIONS

It is important to note that most articles use the term personalized and adaptive systems interchangeably [LNN⁺10]. However, the main difference between personalized and adaptive systems is the fact that adaptive systems take into account the evolution of the user profile over time. In other words, they make use of profile dynamics.

2.4 Conclusions

This article has set forth the state of the art of user profiling and its dynamics. Also, it explains what a user profile is, describes the process of constructing a user profile, and exposes the various ways that it can be used, so the information stored in the user profile changes with the interests and belief of the user, making the profile dynamic. It also defines and exemplifies RS and Personalized/Adaptive Systems where user profiles are of utmost importance to their performance.

The word cloud depicted in Figure 2.6 was generated based on the corpus of extracted papers. This visual representation illustrates the terms most commonly employed for user profiling. The size of each word in the word cloud is directly proportional to the frequency with which it appears in the extracted papers.

As depicted in the figure, the terms “recommendation” and “recommendation systems” are frequently utilized within the domain of user profiling, implying that this particular system serves as the principal application of user profiling. Inside the topic of recommendation systems, a portion of the existing literature proposes methodologies for recommendations that dynamically adapt to the contextual environment in which the user is inserted.

Furthermore, the most relevant information for constructing a user profile is, as per the extracted literature, the user’s interests and preferences. This information can then be represented in the user profile as ontologies, encapsulating knowledge about the relationships among various features.

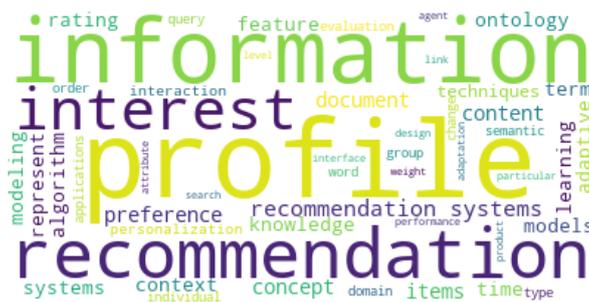


Figure 2.6: Word cloud showing the most frequently used terms in the extracted papers.

Chapter 3

Implementation

The primary objective of this dissertation was to formulate an algorithm for the dynamic creation of user profiles. This algorithm was constructed using Python on the Visual Studio Code platform. This chapter elucidates the developmental trajectory of the algorithm, unveiling the devised architecture for the kb. It delineates the components and relationships of this architecture.

Subsequently, the algorithmic structure is expounded, elaborating on the principal modules and their associated sub-modules. It provides insights into their components, objectives, and internal mechanisms. Lastly, the use cases in which this dissertation's framework will be applied are explicated. Instances of the kb structure for each of these scenarios are presented, accompanied by explanations of the roles fulfilled by various nodes within the kb for each distinct application.

3.1 Tools used

This section presents an overview of the tools employed in the project's implementation. A concise introduction to each tool will be provided, outlining its purpose and functionalities. Furthermore, the rationale behind the selection of these specific tools will be presented.

Python

Python, introduced by Guido van Rossum in the late 1980s, is a high-level programming language with broad applicability across various domains. Its general-purpose nature enhances its versatility, appealing to developers from diverse backgrounds. The language places emphasis on simplicity and readability, ensuring ease of learning. Furthermore, Python boasts a rich collection of libraries and frameworks, catering to a wide range of requirements. Its extensive user community ensures robust support, aiding programmers in overcoming challenges effectively [Ven03].

The decision to utilize Python for the implementation of the user profiling algorithm is grounded in its extensive repository of open-source libraries. These libraries play a pivotal role in simplifying the complex data analysis and processing tasks necessary for this project.

3.2. DEVELOPMENT PHASE

Python's adeptness in scalable data processing solutions enables it to efficiently manage substantial volumes of data, rendering certain Big Data operations feasible within reasonable timeframes. Python's efficiency in dealing with Big Data is underscored by its capacity to yield comparable outcomes with fewer lines of code when juxtaposed with alternative programming languages. Furthermore, Python boasts a diverse assortment of communities comprising individuals hailing from varied backgrounds and fields of expertise. This multiplicity of perspectives and varying levels of experience within these communities fosters the exchange of knowledge and new ideas [RPN20].

Main libraries

There are many libraries that were used in the development of this algorithm, the most notable where NumPy, pandas and scikit-Learn.

NumPy is a Python library that finds application in scientific computing. It introduces a multidimensional array object known as *ndarray*. These arrays serve as a swift and efficient medium for storing, manipulating, and transporting numerical data. Additionally, NumPy furnishes functionalities that facilitate mathematical operations involving arrays, along with enabling internal computations among the elements of an array. This capability streamlines and expedites array processing within Python through the utilization of this library. NumPy also presents an array of linear algebra operations, Fourier transform functions, and the capacity to generate random numbers [McK12].

pandas is a library that offers users a collection of robust data structures along with functions aimed at simplifying the manipulation of structured data. This library streamlines the process of working with structured data by enhancing speed, ease, and expressiveness. It stands out because it combines NumPy's array processing capabilities with the adaptable data manipulation functions akin to those present in spreadsheets and relational databases such as SQL. Moreover, pandas introduces an advanced indexing functionality, which facilitates the execution of data-related operations like slicing, aggregations, dicing, and the selection of data subsets [McK12].

Scikit-Learn is a library utilized for the implementation of diverse machine learning algorithms. It has been designed for individuals who are not necessarily specialized in software development but require the capability to execute statistical data analysis tasks. Consequently, the library prioritizes user-friendliness, performance, robust documentation, and consistent APIs. It is distributed under the BSD license, enabling its integration in both academic and commercial environments [PVG⁺11].

3.2 Development phase

During the initial phases of the project, the KBAI-SE team established what would be the structure of the kb, it was determined that it would be comprised of five main components: Customer, Interest, Item, Variable and Characteristic.

The Customer would be defined as each of the individual users interacting with the system, their demographic information would be recorded enable the creation of demographic

3.2. DEVELOPMENT PHASE

clusters, aiding in inferring information about their preferences. Furthermore, data pertaining to the level of preference Customers held for Interests and Items would also be stored.

The Interests would constitute of a set of categories for which Customers could exhibit varying degrees of interest, or which could be employed to categorize Items. These categories would be organized hierarchically into supercategories and subcategories. Notably, certain interests would be interconnected by specific degrees of similarity. To represent this affinity, a numerical value between 0 and 1 would be designated, with values exceeding 0.75 denoting a significant similarity. Figure 3.1 represents an example of the interest profile of a customer with the hierarchies and affinities represented as well as the preference weights of the customer associated to each interest.

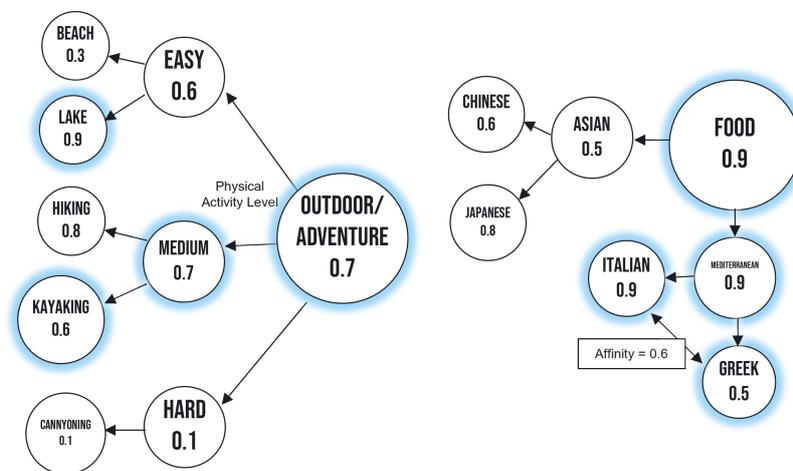


Figure 3.1: Example of an interest profile

The Items would encompass all of the objects that could be recommended to Customers. They would be categorized by establishing relationships with the Interests. Similarly to the interests, Items also can be interconnected by specific degrees of affinity, represented by a numerical value between 0 and 1 would be designated, with values exceeding 0.75 denoting a significant similarity. It also imposes specific conditions for recommending an Item to a Customer by establishing relationships with Characteristics. Figure 3.2 represents an example of the item profile of a customer with the affinities represented as well as the preference weights of the customer associated to each item.

The Variables would represent all of the factors capable of influencing a given session, such as weather for tourism activity recommendations.

The Characteristics would encapsulate the conditions that have to be met for a user to be suggested a particular Item.

Afterwards, the KBAI-SE team established fundamental requisites for the user profiling algorithm.

Firstly, the algorithm needed to operate inconspicuously, gathering user information in the background, while also offering users the option to manually input data through a graphical interface.

3.2. DEVELOPMENT PHASE

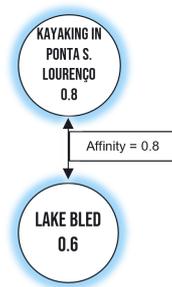


Figure 3.2: Example of an item profile

One of the system's functions would involve recommending Items to Customers based on their level of preference for Interests and Items as well as what their specific queries to the system and the prevailing context. To achieve this, adjustments to customer preferences would be necessary and items would have to be filtered according to the session context, utilizing the information from associated Variables. Items would also need to be filtered based on the customer's characteristics as well as interests and items vetoed by the users. It was decided that vetoes would be represented by the value of -1.

The representation of items and their corresponding interests was determined using an array composed of zeros and ones. In this representation, if an item is linked to a specific interest, it would be denoted by the value one; conversely, if the item lacks an association with the interest, it would be indicated by the value zero. On the other hand, the user's interest profile was envisaged to possess values ranging from 0 to 1, while a value of -1 would signify a veto associated with each of the potential interests, as depicted in fig. 3.3.

After the filtration stage, it was determined that item recommendations would rely on their congruence with the user's preference profile. This alignment would be determined by calculating the proximity between the user's preference profile and the categories associated with the items. As a result, items that exhibit the highest similarity to the user's preference profile would be recommended first, while those with the lowest similarity would be recommended last.

Recognizing the dynamic nature of user interests and preferences, it was imperative for the system to adapt alongside users. Consequently, the team identified that historical user interactions, such as previous search sessions, chosen items, and provided feedback, would play a pivotal role in determining shifts in user interests. To address this, the team devised a periodic analysis of user sessions, along with the items suggested and selected in each session. Of particular significance was assessing whether the user's selections aligned with the top-recommended items for a given session. If a user chose an item that was suggested in a subsequent position, this incongruity would alert the system to potential disparities between the user's profile and their genuine preferences.

To uncover the cause of such disparities, the team devised a comparative analysis of the selected item's categories against those of the first-recommended item. This analysis sought to identify differences in category composition between the two items. The algorithm would document the characteristics unique to the selected item and absent in the first-recommended

3.2. DEVELOPMENT PHASE

	Hiking	Kayaking	Canyoning	
User	0.7	0.5	0.9	
Levada do Moinho	1	0	0	2nd
Canyoning in Ribeira das Cales	1	0	1	1st
Kayaking in Ponta de São Lourenço	0	1	0	3rd

Figure 3.3: Recommendation process example

item, subsequently elevating the values of those corresponding categories. Conversely, the algorithm would decrease the values of categories found in the first-recommended item but absent in the selected item. This process aimed to rectify discrepancies and better align the user profile with their actual preferences (fig. 3.4).

	Item	Char 1	Char 2	Char 3	
Over-ranked	Item 1	Food	Mediterranean	Greek	2
	Item 2	Food	Asian	Japanese	4
	Item 3	Food	Asian	Chinese	3
	Item 4	Food	Mediterranean	Italian	1

Selected →

Figure 3.4: Over-ranked items example

The user profiling algorithm is designed to interact collaboratively with several other modules within the project. These modules encompass the chatbot, the kb API, the system’s graphical interface, and the aggregation module.

The process of collecting both the user’s demographic information and session-related data occurs through the chatbot. As users engage with the chatbot, their interactions generate information that is then transmitted to the user profiling algorithm via the kb API. Additionally, users have the option to interact with the system’s graphical interface to modify their demographic details and preference settings stored within their profile.

Concurrently, the aggregation module undertakes the task of forming user groups and aggregating their respective preferences. The resultant aggregated preferences are subsequently transmitted to the user profiling algorithm via the kb API. This facilitates the algorithm's capability to provide recommendations tailored to the preferences of aggregated groups.

3.3 General structure

The primary output of the KBAI-SE project is the Knowledge Base, a repository where all project-related information is stored and processed, serving as a resource for other modules within the project. To facilitate the development of the kb, the necessity for a comprehensive knowledge ontology structure emerged. However, due to confidentiality reasons, the ontology and its accompanying description cannot be provided.

3.4 Use cases

This dissertation will be applied to three distinct use cases. The initial scenario involves recommending tourism activities to users predicated on their geographic location and specific interests, encompassing nature-oriented activities like kayaking and hiking, entertainment-based options such as museums and amusement parks, and noteworthy events like concerts.

The second application pertains to the recommendation of news articles aligned with users' preferences for specific news categories such as business, entertainment, environment, science, and more.

The third utilization scenario centers on suggesting project tasks to team members contingent on their current workload and expertise.

3.4.1 Tourism

This project works on a case study in the tourism domain, where customers are represented as tourists. The interests of the tourists can be classified into two types: categories of tourism activities and locations of places. The items in this context refer to specific tourism activities.

The characteristics of the tourists include their physical ability or capacity required to engage in an activity, their age group, and whether the activity aligns with their budgetary constraints.

The variables that influence the tourism experience are the weather conditions, which can impact the interests positively or negatively, and the occurrence of an injury, which negatively affects the tourist's physical ability.

In the example of a tourism session, the tourist initially queries about the weather conditions. After receiving the response from the kb, the tourist proceeds with a second query seeking recommendations for activities to do the following day. The system then suggests activities based on the variable of the query, taking into account the specific circumstances of the tourist.

3.5. SYSTEM STRUCTURE

Additionally, there's contextual client-to-self connection, representing a group of users who wish to plan their activities together.

3.4.2 News

A case study in the field of news was also studied, where the clients are represented as readers. The interests of the readers can be classified into two types: categories of news and the newspapers where the news is published. The items, in this context, refer to individual news articles.

The features associated with the readers include the languages in which they comprehend news or the language in which the news articles are written.

The variables that influence the news experience encompass events that increase the popularity of certain news categories (e.g., the Miss Universe pageant making fashion news more popular) and the recent view count of a particular news item (click-through rate).

In this case study, a group of users could be utilized, for instance, to create a regular newsletter that consolidates the interests of the target users, providing them with curated news content tailored to their preferences.

3.4.3 Project Management

Finally, a case study in the field of project management was developed, where the clients are represented as members of a company. The interests of the clients can be classified into two types: task categories and projects. The items in this context refer to the specific tasks that need to be accomplished within the projects.

The characteristics associated with the clients are the knowledge capabilities required to perform the tasks effectively. The variables in this case study are defined by the priority level assigned to each task, indicating its relative importance or urgency.

A group of users in this scenario would represent a team within the company. The aggregation engine, or recommendation system, would then suggest the tasks that align with the abilities and expertise of the group members. This helps ensure that the tasks assigned to the team are suitable for their skill set, increasing efficiency and productivity in project execution.

3.5 System structure

The system is structured around four primary modules, represented in fig. 3.5:

- The first module, referred to as the Explicit Collection Module, enables users to manually furnish the system with information regarding their interests and preferences through the system's interface.
- The second module, known as the Implicit Collection Module, operates in the background, gathering information on users' preferences through their interactions with the

3.5. SYSTEM STRUCTURE

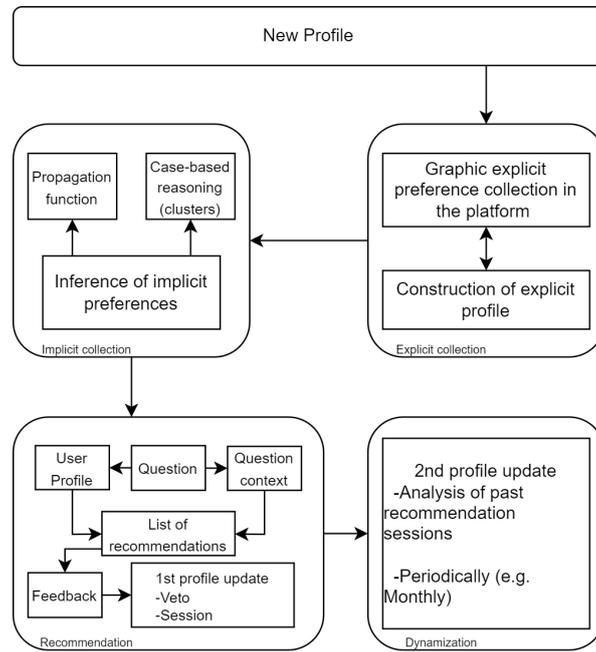


Figure 3.5: System structure

system. This is achieved using a set of algorithms developed in this dissertation, the demographic clustering algorithm and the propagation function.

- The third module, the Recommendation Module, is tasked with the selection of items that closely match a user's preferences within a particular session. This selection process takes into account both the session's context and the specific queries posed by the user to the system. Importantly, users will have the option to provide feedback on the list of recommendations. This feedback mechanism facilitates the automatic update of their preference values within the user profile.
- The fourth module, referred to as the Dynamization Module, is designed to progressively modify the user's preference levels over time. This dynamic adjustment is executed by analyzing the user's historical interactions with the system and evaluating past recommendation sessions. Such evaluation includes considering the feedback provided by the user in response to specific recommendation sessions.

In the subsequent sections, a comprehensive exploration of each module will be conducted, providing an intricate description of their respective functionalities and components. Additionally, the rationale behind the design and development of each module will be elucidated, drawing from the insights garnered through the research conducted in the State-of-the-Art chapter.

3.5.1 Explicit Collection Module

In this phase, the objective is to gather the level of preference values expressed by a customer towards a specific interest or item. These values are collected through graphical interactions within the platform and are used to construct the explicit user profile [IFO15]. Whenever the customer modifies a value within the platform, these preference values are updated accordingly.

The graphical platform consists of four buttons, arranged from left to right, as depicted in Figure 3.6. The functionality of each button is as follows:

1. Veto button: This button assigns an explicit value of -1, indicating strong disapproval or rejection of the interest or item.
2. Dislike button: This button assigns an explicit value of 0.25, indicating a moderate level of dislike towards the interest or item.
3. Like button: This button assigns an explicit value of 0.65, indicating a moderate level of liking or preference towards the interest or item.
4. Love button: This button assigns an explicit value of 0.9, indicating a strong affection or preference towards the interest or item.

By utilizing these buttons, customers can easily express their level of preference, allowing the system to gather valuable information for constructing their explicit user profile.

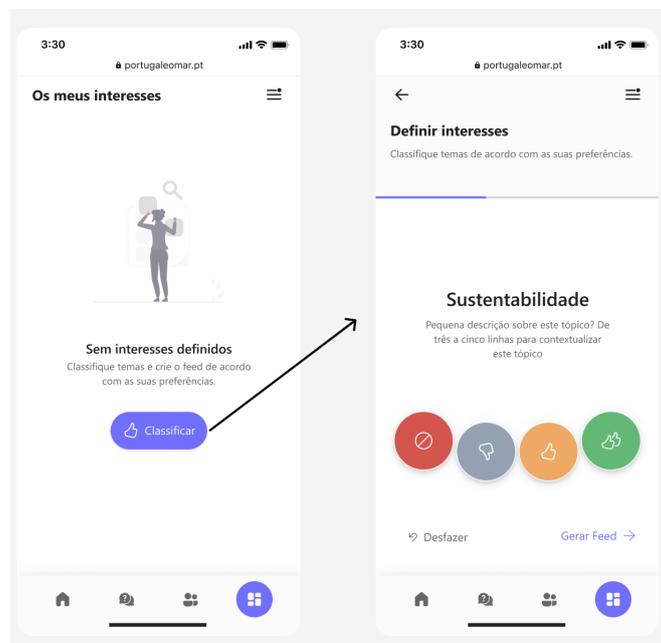


Figure 3.6: Explicit collection platform

3.5.2 Implicit Collection Module

The implicit preference values of the customer towards an interest or item will be inferred by the use of a propagation function and demographic clustering.

Propagation module The propagation module for the interests, located in the file *calculator.py*, is responsible for propagating explicit values collected in the previous phase to parent interests, sub-interests, and affinity interests. This module is utilized whenever a customer modifies the explicit value of an interest in the graphical platform:

- Listing A.1 receives the customer profile, the ID of the interest whose weight will be modified (Int_m) and the new weight value, afterwards, it updates the explicit weight of Int_m (Ew_m) to the new value while preserving the original value ($Ew_{original}$).
- The implicit weight of Int_m (Iw_m) is then modified according to the following propagation steps (listing A.2):
 - If Iw_m is zero, it's value is updated to be the same as the explicit interest ($Iw_m = Ew_m$).
 - If Iw_m is not 0 then Iw_m is set to the mean between Iw_m and Ew_m (listing A.3).
- Listing A.4 modifies the implicit weights of each parent and child interest of Int_m ($Int_{p/c}$) based on the following rules:
 - If both the explicit weight ($Ew_{p/c}$) and implicit weight ($Iw_{p/c}$) of $Int_{p/c}$ are zero, $Iw_{p/c}$ is set to Ew_m
 - If $Ew_{p/c}$ is not zero and $Iw_{p/c}$ is zero, $Iw_{p/c}$ is determined as the mean between $Ew_{p/c}$ and Ew_m (listing A.3).
 - If $Iw_{p/c}$ is not zero, $Iw_{p/c}$ is adjusted as the mean between $Iw_{p/c}$ and Ew_m (listing A.3).
- Listing A.5 modifies the implicit weights of each affinity interest of Int_m (Int_a) according to the following guidelines:
 - If the affinity weight (Aff_w) is greater than 0.75, the implicit weight of Int_a (Iw_a) is modified.
 - The new value of Iw_a is calculated as follows: $\min[(Ew_a + ((Ew_m - Ew_{original}) * Aff_w)), Ew_m]$

Similarly, the propagation module for items in the *calculator.py* file employs explicit values for the items and propagates them to the implicit weight value:

- Listing A.6 receives the customer profile, the ID of the item whose weight will be modified ($Item_m$) and the new weight value. It updates the explicit weight of $Item_m$ (Ew_m) to the new value while saving the original value ($Ew_{original}$).

3.5. SYSTEM STRUCTURE

- Listing A.2 function modifies the implicit weight of Item_m (Iw_m) based on the following steps:
 - If Iw_m is zero, it's value will be updated to be the same as the explicit interest ($Iw_m = Ew_m$).
 - If Iw_m is not zero, Iw_m is set as the mean between Iw_m and Ew_m .

Demographic clustering module The demographic clustering technique is utilized to estimate preference values based on the average preferences of the demographic cluster associated with each user. The following paragraphs describe the clustering module based on the method described in article [Kru97], which is located in the file *clustering.py*:

- To establish a demographic data set for each use case, listing A.7 requires a label list containing the names of the labels to be used (options include: "Age," "Gender," "Country residence," "Education status," "Occupation"). These label values are then encoded and normalized to ensure the algorithm interprets the features consistently (listing A.8, listing A.9).
- To identify the demographic clusters, the k-means clustering method is employed, where the value of k corresponds to the desired number of demographic clusters. This unsupervised machine learning clustering algorithm groups data points into clusters based on similarities within the feature set. As stated in [MVC14], k-means clustering is widely used for this type of analysis.
- The k-means function (listing A.10) receives the desired number of clusters ($n_{clusters}$), the number of times the algorithm will be run with different centroid seeds (n_{init}) and the demographic data set. Finally, it assigns a cluster number to each user.
- Once each user is assigned to a cluster, preference levels can be inferred from their comprehensive profile. This is achieved by calculating the average preference level for each interest category across all users assigned to each cluster (listing A.11, listing A.12).
- If a user has a missing implicit value for a specific interest category, that value is filled with the mean interest value of the corresponding cluster (listing A.13).

3.5.3 Recommendation Module

The recommender module, located in the file *recommender.py*, utilizes the user profile and session variable information to provide item recommendations to the user. The items are filtered based on the user profile and the session variable, making this recommender context aware similarly to the one described in article [XZZG10]. The resulting list of recommendations is subject to user feedback, which is saved in the session information. If a user vetoes an item, this information is updated.

The recommendation process follows these steps:

3.5. SYSTEM STRUCTURE

- Listing B.1 receives the session information, customer profile, a list of items to be recommended, an array containing all the interest categories, and the desired number of recommendations (n).
- The customer profile is modified based on the session variable (listing B.2):
 - The user’s characteristics are adjusted according to the session variable (listing B.3). For example, if the user has the characteristic "Physically Active" with a value of 3 (High), and the session has a variable of "Injury" with a weight of -2 on "Physically Active," then the user’s characteristic for this session will be "Physically Active" with a value of 1 (Low).
 - The user’s preference weights for interests (listing B.4) and items (listing B.5) are also modified based on the session variable. If the interest/item has an explicit weight, it is adjusted by adding the variable weight to it. Otherwise, the implicit weight is modified.
- The item dataset is filtered based on the modified user profile (listing B.6):
 - Items are filtered based on the user’s characteristics (listing B.7). For example, a user with the "Physically Active" characteristic equal to 1 will have all items filtered that have a "Physically Active" characteristic greater than 1.
 - Items associated with vetoed interests are filtered (listing B.8). This includes items with associated interests with an explicit or implicit weight of -1.
 - Vetoed items are also filtered (listing B.9). This includes items where the customer profile has an explicit or implicit weight of -1 for the item.
- The distance between the filtered items and the user profile is measured through listing B.10, and the recommender returns the n closest items to the customer profile (listing B.11, listing B.12).

3.5.4 Dynamization Module

The dynamization module, located in the file *calculator.py*, utilizes information from previous sessions to gradually modify profile preference values through the periodic analysis of feedback collected during recommendation sessions, similarly to what is described in article [FGD⁺23]. Interests that require an increase or decrease in weight are identified and adjusted based on the following set of rules:

- Listing C.1 receives the session array, customer profile, interest categories, the number of positions that an item must go up (n_{up}) and down (n_{down}) after analyzing the user feedback (i.e. if n_{up} is equal to two then the difference between the recommended position and the final position of the item in the session has to be greater than two for the item to be analyzed) and finally the date indicating when the sessions should start being analyzed (i.e. if the date is 25/01/2023 then only the sessions that occurred after that date will be analyzed).

3.5. SYSTEM STRUCTURE

- For each session, the recommended items (I_n) are analyzed (listing C.2):
 - If the difference between the recommended position and the final position is greater than n_{up} , the first recommended item (I_{r1}) is saved, and all interests associated to I_n are analyzed. For each interest category (Int_{cat}), the following conditions are checked (listing C.3):
 - * If Int_{cat} is not associated with I_{r1} but is associated with I_n , the implicit interest value of Int_{cat} is saved in an array (Arr_{up}) to track interests whose value will increase.
 - * If Int_{cat} is associated with I_{r1} but not with I_n , the implicit interest value of Int_{cat} is saved in an array (Arr_{down}) to track interests whose value will decrease.
 - If the difference between the recommended position and the final position is less than n_{down} , the first item chosen by the user (I_{f1}) is saved, and all interests associated with I_n are analyzed. For each interest category (Int_{cat}) the following conditions are checked (listing C.3):
 - * If Int_{cat} is not associated with I_{f1} but is associated with I_n , the implicit interest value of Int_{cat} will be saved in an array (Arr_{down}) with all the Int_{cat} to track interests whose value will decrease.
 - * If Int_{cat} is associated with I_{f1} but not with I_n then the implicit interest value of Int_{cat} will be saved in an array (Arr_{up}) to track interests whose value will increase.
- The mean number of positions (m_{pos}) that an Int_{cat} moved up or down when associated with an item is saved (listing C.4).
- The frequency of occurrence of each interest category in Arr_{up} and Arr_{down} ($freq_{arr}$) is calculated and saved (listing C.4).
- The implicit value of each Int_{cat} (Iw_{cat}) is calculated by adding or subtracting (depending on whether the category is in Arr_{up} or Arr_{down} respectively) the result of: $Iw_{cat} * (m_{pos} + freq_{arr})/100$ (listing C.5)

Figure 3.7 shows an example of the dynamization module being applied on three interest categories, it shows that, when the module is applied on a category, the changes it applies on it's values are small, this occurs to ensure that it's aligned with Belief Revision principles and that no abrupt changes are made.

3.5. SYSTEM STRUCTURE

Client id: 42	Result:
Day/month/year: 25/02/2023	
Number of positions = 2	$IW_{italian} = 0.5 + \frac{0.5*(3+1)}{100} = 0.52$
Interests up = Italian ($m_{pos} = 3, freq_{arr} = 1, IW_{cat} = 0.5$)	$IW_{asian} = 0.8 - \frac{0.8*(2+1)}{100} = 0.776$
Interests down = Asian ($m_{pos} = -2, freq_{arr} = 1, IW_{cat} = 0.8$), Japanese ($m_{pos} = -2, freq_{arr} = 1, IW_{cat} = 0.7$)	$IW_{japanese} = 0.7 - \frac{0.7*(2+1)}{100} = 0.679$
To each interest category implicit weight add or subtract:	
$\frac{IW_{cat} * (m_{pos} + freq_{arr})}{100}$	

Figure 3.7: Dynamization module example

Chapter 4

Conclusions and Future Work

The dissertation was initiated by introducing the KBAI-SE project, elucidating its conceptual underpinnings, objectives, and overarching framework. Subsequently, it delved into the delineation of the project's specific scope relevant to this dissertation. This pertained to the formulation and management of user profiles intended for utilization within the KBAI-SE knowledge base. Notably, the challenges and complexities inherent in the process of user modeling were underscored. This was driven by the imperative to adeptly tailor information recommendations to users based on their distinct interests and requirements.

Furthermore, the dissertation offered an initial blueprint for the user modeling procedure. It illuminated the practical contexts in which the project would be situated, concurrently elucidating the preliminary stages mapped out for the construction of the user profile algorithm. This algorithm would be elaborated upon in subsequent chapters.

It followed with the review of the current state-of-the-art in the realm of user profiling techniques and their dynamics. It compiled a comprehensive collection of various potential techniques employed for constructing user profiles. This encompassed methodologies for gathering user-related data, extracting user features, depicting user preference levels, and discerning alterations in user preferences over time.

Moreover, it also elucidated the diversity of systems where user profiles find utility. This presentation encompassed an exposition of the distinct categories of systems that employ user profiles, while also highlighting the strengths and weaknesses associated with each category.

Finally, the dissertation showcases the execution of the user profiling algorithm devised within this project. It outlines the fundamental architecture of the kb, expounding upon the diverse nodes constituting its framework, alongside their mutual interconnections. The system's structural dynamics are explicated, with particular attention directed towards the elucidation of its core components - four primary modules - as well as their respective sub-modules. Each module's distinct role and functionalities are meticulously delineated.

Moreover, the dissertation encompasses an exposition of three real-world application scenarios that the project is poised to engage with. These scenarios are thoroughly detailed, replete with illustrative examples embedded within the kb for each of the specified use cases.

4.1 Main contributions

The article forming the foundation of the state-of-the-art in this dissertation outlined the essential stages for constructing a user profile. Additionally, it introduced a novel approach to categorize the diverse techniques employed within each stage to model user profiles. It also comprehensively gathered various methods used for incorporating profile dynamics. The end result was a meticulous portrayal of the dynamic user profiling process, delineating the necessary phases for constructing a user profile and ensuring its adaptability to user evolution.

Furthermore, it devised a dynamic user profiling algorithm founded on Belief Revision principles. This algorithm possesses the capacity to deduce user preferences from user interactions with the system, coupled with explicitly provided information. Moreover, it exhibits the capability to identify shifts in user interests over time and progressively adjust the user preference profile to accurately mirror these shifts.

4.2 Future work

To continue this work, several proposed steps are recommended. Firstly, it is suggested to conduct comprehensive efficiency tests on the user profiling algorithm. Additionally, experimentation with alternative algorithms should be undertaken to determine if there are more effective approaches for inferring preferences and detecting changes in user interests. This entails a thorough comparison of various algorithms, leading to the identification of the most efficient one.

Moreover, the exploration of alternative user profiling techniques, such as machine learning-based or ontology-based methods, is advised. This exploration should involve a meticulous analysis of the advantages and limitations of each technique, leading to the selection of the most promising approaches.

Finally, a more profound investigation into the various user profiling techniques for preference learning that were outlined in the state-of-the-art should be conducted. This involves a detailed examination of these techniques, elaborating on their strengths and weaknesses. Furthermore, providing examples of user profiling systems that utilize these techniques would contribute to a more comprehensive understanding of their real-world applications.

Bibliography

- [AAO14] Farida Achemoukh and Rachid Ahmed-Ouamer. Representation and evolution of user profile in information retrieval based on bayesian approach. In International Symposium on Methodologies for Intelligent Systems, pages 486–492. Springer, 2014.
- [AGM85] Carlos Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. Journal of Symbolic Logic, 50:510–530, 1985.
- [AHX13] Ahmad Abdel-Hafez and Yue Xu. A survey of user modelling in social media websites. Computer and Information Science, 6(4):59–71, 2013.
- [AION15] Bahram Amini, Roliana Ibrahim, Mohd Shahizan Othman, and Mohammad Ali Nematbakhsh. A reference ontology for profiling scholar’s background knowledge in recommender systems. Expert Systems with Applications, 42(2):913–928, 2015.
- [AS99] Giuseppe Amato and Umberto Straccia. User profile modeling and applications to digital libraries. In International conference on theory and practice of digital libraries, pages 184–197. Springer, 1999.
- [AT05] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering, 17(6):734–749, 2005.
- [BBC⁺08] Paolo Boldi, Francesco Bonchi, Carlos Castillo, Debora Donato, Aristides Gionis, and Sebastiano Vigna. The query-flow graph: model and applications. In Proceedings of the 17th ACM conference on Information and knowledge management, pages 609–618, 2008.
- [BHBTC22] Aymen Ben Hassen, Sonia Ben Ticha, and Anja Habacha Chaibi. Deep learning for visual-features extraction based personalized user modeling. SN Computer Science, 3(4):261, 2022.

- [BKN07] Peter Brusilovski, Alfred Kobsa, and Wolfgang Nejdl. The adaptive web: methods and strategies of web personalization, volume 4321. Springer Science & Business Media, 2007.
- [BL07] James Bennett and Stan Lanning. The netflix prize. 2007.
- [BOHG13] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. Knowledge-based systems, 46:109–132, 2013.
- [Bro16] Dermot Browne. Adaptive user interfaces. Elsevier, 2016.
- [BWC⁺12] Paul N Bennett, Ryen W White, Wei Chu, Susan T Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. Modeling the impact of short- and long-term behavior on search personalization. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, pages 185–194, 2012.
- [CC12] Victor Codina and Luigi Ceccaroni. Semantically-enhanced recommenders. In CCIA, pages 69–78, 2012.
- [CKL15] Lun-Chi Chen, Ping-Jen Kuo, and I-En Liao. Ontology-based library recommender system using mapreduce. Cluster Computing, 18(1):113–121, 2015.
- [CLS⁺21] Hongxu Chen, Yicong Li, Xiangguo Sun, Guandong Xu, and Hongzhi Yin. Temporal meta-path guided explainable recommendation. In Proceedings of the 14th ACM international conference on web search and data mining, pages 1056–1064, 2021.
- [CMVGRG⁺15] Luis Omar Colombo-Mendoza, Rafael Valencia-García, Alejandro Rodríguez-González, Giner Alor-Hernández, and José Javier Samper-Zapater. Recommetz: A context-aware knowledge-based mobile recommender system for movie showtimes. Expert Systems with Applications, 42(3):1202–1222, 2015.
- [CP09] Wei Chu and Seung-Taek Park. Personalized recommendation on dynamic content using predictive bilinear models. In Proceedings of the 18th international conference on World wide web, pages 691–700, 2009.
- [CZH⁺17] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17, page 335–344, New York, NY, USA, 2017. Association for Computing Machinery.

- [DSRO20] Aminu Da’u, Naomie Salim, Idris Rabi, and Akram Osman. Weighted aspect-based opinion mining using deep learning for recommender system. Expert Systems with Applications, 140:112871, 2020.
- [DZD22] Chenlong Deng, Yujia Zhou, and Zhicheng Dou. Improving personalized search with dual-feedback network. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, pages 210–218, 2022.
- [EHNA19] Boulkrinat Nour El Houda, Benblidia Nadjia, and Meziane Abdelkrim. Queries-based profile evolution using genetic algorithm. In 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), pages 1–6. IEEE, 2019.
- [ENSN19] Christopher Ifeanyi Eke, Azah Anir Norman, Liyana Shuib, and Henry Friday Nweke. A survey of user profiling: State-of-the-art, challenges, and solutions. IEEE Access, 7:144907–144924, 2019.
- [FEMR18] Marina Farid, Rania Elgohary, Ibrahim Moawad, and Mohamed Roushdy. User profiling approaches, modeling, and personalization. In Proceedings of the 11th International Conference on Informatics & Systems (INFOS 2018), 2018.
- [FFJZ06] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, and Markus Zanker. An integrated environment for the development of knowledge-based recommender applications. International Journal of Electronic Commerce, 11(2):11–34, 2006.
- [FGD⁺23] Eduardo Fermé, Marco Garapa, Maurício D.L., Reisand Yuri Almeida, Teresa Paulino, and Mariana Rodrigues. Knowledge-driven profile dynamics. Submitted manuscript, 2023.
- [FH18] Eduardo Fermé and Sven Ove Hansson. Belief Change: Introduction and Overview. Springer Briefs in Computer Science Series. Springer, 2018.
- [FL07] Carsten Felden and Markus Linden. Ontology-based user profiling. In International Conference on Business Information Systems, pages 314–327. Springer, 2007.
- [FP06] Haiyan Fan and Marshall Scott Poole. What is personalization? perspectives on the design and implementation of personalization in information systems. Journal of Organizational Computing and Electronic Commerce, 16(3-4):179–202, 2006.

- [FPW⁺19] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. Proceedings of the AAAI Conference on Artificial Intelligence, 33(1):94–101, Jul. 2019.
- [FTDCM18] Golnoosh Farnadi, Jie Tang, Martine De Cock, and Marie-Francine Moens. User profiling through deep multimodal fusion. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pages 171–179, 2018.
- [GLH⁺21] Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. Advances and challenges in conversational recommender systems: A survey. AI Open, 2:100–126, jan 2021.
- [GSCM07] Susan Gauch, Mirco Speretta, Aravind Chandramouli, and Alessandro Micarelli. User profiles for personalized information access. The adaptive web, pages 54–89, 2007.
- [GXCM14] Qian Gao, Su Mei Xi, Young Im Cho, and Eric T Matson. A multi-agent context-based personalized user preference profile construction approach. In Soft Computing in Advanced Robotics, pages 55–69. Springer, 2014.
- [HF15] Ahmad Hawalah and Maria Fasli. Dynamic user profiles for web personalisation. Expert Systems with Applications, 42(5):2547–2569, 2015.
- [HPV16] Chen He, Denis Parra, and Katrien Verbert. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. Expert Systems with Applications, 56:9–27, 2016.
- [IFO15] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adewoke Ojokoh. Recommendation systems: Principles, methods and evaluation. Egyptian informatics journal, 16(3):261–273, 2015.
- [KBV09] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. Computer, 42(8):30–37, 2009.
- [Kru97] Bruce Krulwich. Lifestyle finder: Intelligent user profiling using large-scale demographic data. AI magazine, 18(2):37–37, 1997.
- [KSYF18] Seong She Keng, Cheng Haw Su, Gene Loh Yu, and Fang Chua Fang. Ak tourism: a property graph ontology-based tourism recommender system. 2018.
- [Lau02] Raymond Yk Lau. The state of the art in adaptive information agents. International Journal on Artificial Intelligence Tools, 11(01):19–61, 2002.

- [LCLH19] Shang Liu, Zhenzhong Chen, Hongyi Liu, and Xinghai Hu. User-video co-attention network for personalized micro-video recommendation. In The World Wide Web Conference, WWW '19, page 3020–3026, New York, NY, USA, 2019. Association for Computing Machinery.
- [LJMR09] Víctor López-Jaquero, Francisco Montero, and Fernando Real. Designing user interface adaptation rules with t: Xml. In Proceedings of the 14th international conference on Intelligent user interfaces, pages 383–388, 2009.
- [LL09] Kenneth Wai-Ting Leung and Dik Lun Lee. Deriving concept-based user profiles from search engine logs. IEEE Transactions on knowledge and data engineering, 22(7):969–982, 2009.
- [LMNS19] Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. Semantics in Adaptive and Personalised Systems. Springer, 2019.
- [LN12] Claudia Lauschke and Eirini Ntoutsis. Monitoring user evolution in twitter. In 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pages 972–977. IEEE, 2012.
- [LNN⁺10] Duc-Long Le, An-Te Nguyen, Dinh-Thuc Nguyen, Van-Hao Tran, and Axel Hunger. A survey of applying user profile in the adaptive instructional systems. In Proceedings of the 5th International Conference on ELearning, Penang, Malaysia, 2010.
- [LNNH09] Duc-Long Le, An-Te Nguyen, Dinh-Thuc Nguyen, and Axel Hunger. Building learner profile in adaptive e-learning systems. In Proceedings of the 4th International Conference on e-Learning (ICEL 2009), pages 15–17, 2009.
- [LPL⁺16] Zhongqi Lu, Sinno Jialin Pan, Yong Li, Jie Jiang, and Qiang Yang. Collaborative evolution for user profiling in recommender systems. In IJCAI, pages 3804–3810, 2016.
- [LS12] Raymond Y. K. Lau and Long Song. Belief Revision for Intelligent Web Service Recommendation, pages 53–66. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [LYWK07] Lin Li, Zhenglu Yang, Botao Wang, and Masaru Kitsuregawa. Dynamic adaptation strategies for long-term and short-term user profile to personalize search. In Advances in Data and Web Management, pages 228–240. Springer, 2007.
- [LZE15] Dawen Liang, Minshu Zhan, and Daniel PW Ellis. Content-aware collaborative music recommendation using pre-trained neural networks. In ISMIR, pages 295–301, 2015.

- [MAC⁺07] Golemati Maria, Katifori Akrivi, Vassilakis Costas, L George, and Halatsis Constantin. Creating an ontology for the user profile: Method and applications. In Proceedings AI* AI Workshop RCIS, 2007.
- [McK12] Wes McKinney. Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. " O'Reilly Media, Inc.", 2012.
- [MGA⁺15] Mahsa Mehrpoor, Jon Atle Gulla, Dirk Ahlers, Kjetil Kristensen, Soroush Ghodrat, and Ole Ivar Sivertsen. Using process ontologies to contextualize recommender systems in engineering projects for knowledge access improvement. Academic Conferences and Publishing International, 2015.
- [MIM10] Lucas Marin, David Isern, and Antonio Moreno. A generic user profile adaptation framework. In Artificial Intelligence Research and Development, pages 143–152. IOS Press, 2010.
- [MIM13] Lucas Marin, David Isern, and Antonio Moreno. Dynamic adaptation of numerical attributes in a user profile. Applied intelligence, 39(2):421–437, 2013.
- [MMI14] Lucas Marin, Antonio Moreno, and David Isern. Automatic preference learning on numeric and multi-valued categorical attributes. Knowledge-Based Systems, 56:201–215, 2014.
- [Mou97] Alexandros Moukas. User modeling in a multiagent evolving system. In Proceedings, workshop on Machine Learning for User Modeling, 6 th International Conference on User Modeling, Chia Laguna, Sardinia. Cite-seer, 1997.
- [MVC14] Héctor D Menéndez, Rafael Vindel, and David Camacho. Combining time series and clustering to extract gamer profile evolution. In International Conference on Computational Collective Intelligence, pages 262–271. Springer, 2014.
- [OLEKC18] Charbel Obeid, Inaya Lahoud, Hicham El Khoury, and Pierre-Antoine Champin. Ontology-based recommender system in higher education. In Companion Proceedings of the The Web Conference 2018, pages 1031–1034, 2018.
- [Puk14] Noppamas Pukkhem. Lorecommendnet: an ontology-based representation of learning object recommendation. In Recent Advances in Information and Communication Technology, pages 293–303. Springer, 2014.
- [PVG⁺11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. the Journal of machine Learning research, 12:2825–2830, 2011.

- [RDS⁺21] Logesh Ravi, Malathi Devarajan, Arun Kumar Sangaiah, Lipo Wang, and V Subramaniaswamy. An intelligent location recommender system utilising multi-agent induced cognitive behavioural model. Enterprise Information Systems, 15(10):1376–1394, 2021.
- [RHS⁺13] Tuukka Ruotsalo, Krister Haav, Antony Stoyanov, Sylvain Roche, Elena Fani, Romina Deliai, Eetu Mäkelä, Tomi Kauppinen, and Eero Hyvönen. Smartmuseum: A mobile recommender system for the web of data. Journal of Web Semantics, 20:50–67, 2013.
- [RJ14a] Chhavi Rana and Sanjay Kumar Jain. An evolutionary clustering algorithm based on temporal features for dynamic recommender systems. Swarm and Evolutionary Computation, 14:21–30, 2014.
- [RJ14b] Chhavi Rana and Sanjay Kumar Jain. An extended evolutionary clustering algorithm for an adaptive recommender system. Social Network Analysis and Mining, 4(1):1–13, 2014.
- [RJ15] Chhavi Rana and Sanjay Kumar Jain. A study of the dynamic features of recommender systems. Artificial Intelligence Review, 43(1):141–153, 2015.
- [RL08] Liana Razmerita and Miltiadis D Lytras. Ontology-based user modelling personalization: Analyzing the requirements of a semantic learning portal. In World Summit on Knowledge Society, pages 354–363. Springer, 2008.
- [RPN20] Sebastian Raschka, Joshua Patterson, and Corey Nolet. Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence. Information, 11(4):193, 2020.
- [SA19] Huma Samin and Tayyaba Azim. Knowledge based recommender system for academia using machine learning: A case study on higher education landscape of pakistan. IEEE Access, 7:67081–67093, 2019.
- [SCP12] V Subramaniaswamy and S Chenthur Pandian. Effective tag recommendation system based on topic ontology using wikipedia and wordnet. International journal of intelligent systems, 27(12):1034–1048, 2012.
- [SDD08] Michael Sutterer, Olaf Droegehorn, and Klaus David. User profile selection by means of ontology reasoning. In 2008 Fourth Advanced International Conference on Telecommunications, pages 299–304. IEEE, 2008.
- [SMG06] Robbie Schaefer, Wolfgang Mueller, and Jinghua Groppe. Profile processing and evolution for smart environments. In International Conference on Ubiquitous Intelligence and Computing, pages 746–755. Springer, 2006.

- [TCJ⁺10] Dieudonné Tchunte, Marie-Françoise Canut, Nadine Baptiste Jessel, André Péninou, and Anass El Haddadi. Visualizing the evolution of users' profiles from online social networks. In 2010 International Conference on Advances in Social Networks Analysis and Mining, pages 370–374. IEEE, 2010.
- [THHK21] Chang-You Tai, Liang-Ying Huang, Chien-Kun Huang, and Lun-Wei Ku. User-centric path reasoning towards explainable recommendation. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 879–889, 2021.
- [TM07] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In 2007 IEEE 23rd international conference on data engineering workshop, pages 801–810. IEEE, 2007.
- [TNM18] John K Tarus, Zhendong Niu, and Ghulam Mustafa. Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. Artificial intelligence review, 50(1):21–48, 2018.
- [TNY17] John K. Tarus, Zhendong Niu, and Abdallah Yousif. A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. Future Generation Computer Systems, 72:37–48, 2017.
- [TVB09] Jan Thomsen, Yves Vanrompay, and Yolande Berbers. Evolution of context-aware user profiles. In 2009 International Conference on Ultra Modern Telecommunications & Workshops, pages 1–6. IEEE, 2009.
- [TYZZ10] Jie Tang, Limin Yao, Duo Zhang, and Jing Zhang. A combination approach to web user profiling. ACM Transactions on Knowledge Discovery from Data (TKDD), 5(1):1–44, 2010.
- [UB02] Supiya Ujjin and Peter J Bentley. Learning user preferences using evolution. In Proceedings of the 4th Asia-Pacific conference on simulated evolution and learning, Singapore, 2002.
- [Ven03] Bill Venners. The making of python: A conversation with guido van rossum, part i. Artima Developer (Jan. 2003). url: <http://www.artima.com/intv/pythonP.html>, 2003.
- [WC08] Sung-Shun Weng and Hui-Ling Chang. Using ontology network analysis for research document recommendation. Expert Systems with Applications, 34(3):1857–1869, 2008.
- [WHW⁺22] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. IEEE Transactions on Knowledge and Data Engineering, 2022.

- [WSLM18] Jiqing Wen, James She, Xiaopeng Li, and Hui Mao. Visual background recommendation for dance performances using deep matrix factorization. ACM Trans. Multimedia Comput. Commun. Appl., 14(1), jan 2018.
- [XLW⁺20] Ruobing Xie, Cheng Ling, Yalong Wang, Rui Wang, Feng Xia, and Leyu Lin. Deep feedback network for recommendation. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, pages 2519–2525, Yokohama, Japan, jul 2020. International Joint Conferences on Artificial Intelligence Organization.
- [XYZ⁺10] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal recommendation on graphs via long- and short-term preference fusion. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, page 723–732, New York, NY, USA, 2010. Association for Computing Machinery.
- [XZZG10] Kaijian Xu, Manli Zhu, Daqing Zhang, and Tao Gu. Context-aware content filtering and presentation for pervasive and mobile information systems. In 1st International ICST Conference on Ambient Media and Systems, 2010.
- [YCC⁺15] Hongzhi Yin, Bin Cui, Ling Chen, Zhiting Hu, and Xiaofang Zhou. Dynamic user modeling in social media systems. ACM Transactions on Information Systems (TOIS), 33(3):1–44, 2015.
- [YDH21] Zuoxi Yang, Shoubin Dong, and Jinlong Hu. Gfe: General knowledge enhanced framework for explainable sequential recommendation. Knowledge-Based Systems, 230:107375, 2021.
- [ZDC⁺20] Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, page 179–188, New York, NY, USA, 2020. Association for Computing Machinery.
- [ZLXC16] Zhiyuan Zhang, Yun Liu, Guandong Xu, and Haiqiang Chen. A weighted adaptation method on learning user preference profile. Knowledge-Based Systems, 112:114–126, 2016.
- [ZYL⁺17] Huan Zhao, Quanming Yao, Jianda Li, Yangqiu Song, and Dik Lun Lee. Meta-graph based recommendation fusion over heterogeneous information networks. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, page 635–644, New York, NY, USA, 2017. Association for Computing Machinery.

CHAPTER 4. CONCLUSIONS AND FUTURE WORK

- [ZYST19] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 52(1):1–38, 2019.

Appendix A

Implicit Collection Module

A.1 Propagation module

Listing A.1: Propagation function for interests

```
def propagation_interest_module(customer: Customer, modified_id, final):
    modified=find_in_dictionary(customer.interests, modified_id)
    original= modified["explicit_weight"]
    modified["explicit_weight"] = final
    do_propagation_operation(modified)
    propagate_interests(modified, customer.interests, original)
return customer
```

Listing A.2: Modify value of implicit weight of target interest

```
def do_propagation_operation(object):
    if object['implicit_weight'] == 0:
        object['implicit_weight'] = copy.copy(object['explicit_weight'])
    elif object['implicit_weight'] != 0:
        object['implicit_weight'] = do_algorithm(object['implicit_weight'],
        object['explicit_weight'])
```

Listing A.3: Calculate mean between two values

```
def do_algorithm(value1, value2):
    values = [value1, value2]
    result = statistics.mean(values)
return round(result, 2)
```

Listing A.4: Modify value of implicit weight of parent and child interests of target interest

```
def do_parent_child_propagation_operation(interest_main, interest):
    if interest['explicit_weight'] == 0 and interest['implicit_weight'] == 0:
        interest['implicit_weight'] = copy.copy(interest_main['explicit_weight'])
```

```
elif interest['explicit_weight'] != 0 and interest['implicit_weight'] == 0:
    interest['implicit_weight'] = copy.copy(interest['explicit_weight'])
    interest['implicit_weight'] =
    do_algorithm(interest['implicit_weight'],
    interest_main['explicit_weight'])
elif interest['implicit_weight'] != 0:
    interest['implicit_weight'] =
    do_algorithm(interest['implicit_weight'],
    interest_main['explicit_weight'])
```

Listing A.5: Modify implicit weights of affinity interests

```
def affinity_algorithm(main_original, main_final, affinity_original, weight):
    operation = affinity_original + ((main_final-main_original)*weight)
    values = [operation, main_final]
    result = min(values)
    return round(result, 2)
```

Listing A.6: Update explicit and implicit weight of the item

```
def propagation_item_module(customer:Customer, modified_id, final):
    modified=find_in_dictionary(customer.items, modified_id)
    modified["explicit_weight"] = final
    do_propagation_operation(modified)
    return customer
    return round(result, 2)
```

A.2 Demographic clustering module

Listing A.7: Create demographic clusters and infer preference weights

```
def clustering_module(customer_array, interest_array, num_clusters, labels_list):
    demographic_init_df = customer_data_demographic(customer_array, labels_list)

    interest_categories = interest_category_adder(interest_array)
    interest_df = customer_data_interests(customer_array, interest_categories)

    demographic_df = encode_labels(demographic_init_df.copy(), labels_list)

    normalized_dataset = normalize_values(demographic_df)

    df, demographic_df, interest_df = apply_k_means(normalized_dataset,
    demographic_init_df, interest_df, num_clusters)
```

```
cluster_mean_df = get_mean_interests(interest_df)

final_df = add_mean_weights(interest_df, cluster_mean_df)

customer_array = fill_customer_data(customer_array, interest_categories, final_df)

return customer_array
```

Listing A.8: Encode labels

```
def encode_labels(df, labels_list):

    if "Gender" in labels_list:
        gender = le.fit_transform(df['Gender'])
        df.drop("Gender", axis=1, inplace=True)
        df["Gender"] = gender
    if "Country_residence" in labels_list:
        country_residence = le.fit_transform(df['Country_residence'])
        df.drop("Country_residence", axis=1, inplace=True)
        df["Country_residence"] = country_residence
    if "Education_status" in labels_list:
        education_status = le.fit_transform(df['Education_status'])
        df.drop("Education_status", axis=1, inplace=True)
        df["Education_status"] = education_status
    if "Occupation" in labels_list:
        occupation = le.fit_transform(df['Occupation'])
        df.drop("Occupation", axis=1, inplace=True)
        df["Occupation"] = occupation

    return df
```

Listing A.9: Normalize values

```
def normalize_values(df):
    X = df.values[:,1:]
    X = np.nan_to_num(X)
    normalized_dataset = StandardScaler().fit_transform(X)
    return normalized_dataset
```

Listing A.10: Use k-means to create and assign clusters

```
def apply_k_means(normalized_dataset, demographic_df, interest_df, num_clusters):
    k_means = KMeans(n_clusters=num_clusters, n_init=12)
    k_means.fit(normalized_dataset)
    labels = k_means.labels_
    df = pd.merge(demographic_df, interest_df)
```

```
df["Cluster_number"] = labels
demographic_df["Cluster_number"] = labels
interest_df["Cluster_number"] = labels
col_cluster = interest_df.pop("Cluster_number")
interest_df.insert(0, "Cluster_number", col_cluster)
col_cluster_d = demographic_df.pop("Cluster_number")
demographic_df.insert(0, "Cluster_number", col_cluster_d)
return df, demographic_df, interest_df
```

Listing A.11: Calculate average preference level for each category by cluster

```
def get_mean_interests(interest_df):
    data_customer=[]
    clusters = interest_df["Cluster_number"].unique()
    df = interest_df.copy()
    df.drop("ID", axis=1, inplace=True)
    df.drop("reference", axis=1, inplace=True)
    col_cluster = df.pop("Cluster_number")
    df = df.replace(0, np.NaN)
    df.insert(0, "Cluster_number", col_cluster)
    for cluster in clusters:
        filtered_df =df[df['Cluster_number'] == cluster]
        dicts_record = filtered_df.mean(axis=0).to_dict()
        dicts_record["Cluster_number"] = cluster
        data_customer.append(dicts_record)
    final_df = pd.DataFrame(data_customer)
    col_cluster = final_df.pop("Cluster_number")
    final_df = final_df.replace(np.NaN, 0)
    final_df.insert(0, "Cluster_number", col_cluster)
    return final_df
```

Listing A.12: Add average weights

```
def add_mean_weights(interest_df, cluster_mean_df):
    clusters = interest_df["Cluster_number"].copy().unique()
    int_df = interest_df.copy()
    int_df.drop("reference", axis=1, inplace=True)
    for cluster in clusters:
        filtered_int_df =int_df[int_df['Cluster_number'] == cluster]
        filtered_mean_df =cluster_mean_df[cluster_mean_df['Cluster_number']
        == cluster]
        column_values = filtered_int_df.columns.values.tolist()
        column_values.remove('Cluster_number')
        column_values.remove('ID')
        for i in column_values:
```

```
    mean = float(filtered_mean_df.iloc[0][i])
    filtered_int_df.loc[filtered_int_df[i] == 0, i] = mean
    int_df[int_df['Cluster_number'] == cluster] = filtered_int_df
return int_df
```

Listing A.13: Infer missing preference weights of customers based on the average cluster weights

```
def fill_customer_data(customer_array, interest_categories, interest_df):
    for customer in customer_array:
        customer_row = interest_df.loc[interest_df['ID'] == customer.id]
        for interest_cat in interest_categories:
            for interest_value in interest_cat:
                for ints in customer.interests:
                    if ints['implicit_weight'] == 0 and ints['explicit_weight'] == 0 and
                    ints['interest'].value == interest_value:
                        ints['implicit_weight'] =
                            customer_row.loc[customer_row.index[0], interest_value]
    return customer_array
```

Appendix B

Recommendation module

Listing B.1: Recommender module

```
def recommender_module(session : Session, items, customer, int_categories, n_rec:int):
    start_time = time.time()
    modified_customer = modify_session_variables(copy.deepcopy(customer), session)
    filtered_items = filtering_module(modified_customer, items)
    item_dataset = item_data_creator(filtered_items, modified_customer, int_categories)
    customer_dataset = customer_data_creator(modified_customer, int_categories)
    recommendations = make_customer_recommendation(customer_dataset,
    item_dataset, n_rec)
    recommendation_array=return_recommendations(recommendations, filtered_items)
    session = add_recommendations(session, recommendation_array, modified_customer)
return(session)
```

Listing B.2: Change values based on session context

```
def modify_variable(customer, variable):
    for c in variable.influence_char:
        customer = modify_characteristic(customer, c)
    for i in variable.influence_int:
        customer = modify_interest(customer, i)
    for j in variable.influence_item:
        customer = modify_item(customer, j)
    return customer

def modify_session_variables(customer, session):
    for c in session.variables:
        customer = modify_variable(customer, c['variable'])
    return customer
```

Listing B.3: Modify characteristic values based on session context

```
def modify_characteristic (customer, characteristic):
```

```
for c in customer.characteristics:  
    if c['ID'] == characteristic['ID']:  
        c['characteristic'].value = c['characteristic'].value + characteristic['weight']  
return customer
```

Listing B.4: Modify interest values based on session context

```
def modify_interest(customer, interest):  
    for i in customer.interests:  
        if i['ID'] == interest['ID']:  
            if i['explicit_weight'] != 0:  
                i['explicit_weight'] = i['explicit_weight'] + interest['weight']  
            elif i['implicit_weight'] != 0:  
                i['implicit_weight'] = i['implicit_weight'] + interest['weight']  
    return customer
```

Listing B.5: Modify item values based on session context

```
def modify_item (customer, item):  
    for i in customer.items:  
        if i['ID'] == item['ID']:  
            if i['explicit_weight'] != 0:  
                i['explicit_weight'] = i['explicit_weight'] + item['weight']  
            elif i['implicit_weight'] != 0:  
                i['implicit_weight'] = i['implicit_weight'] + item['weight']  
    return customer
```

Listing B.6: Filtering module

```
def filtering_module(customer: Customer, items):  
    filtered_items = filter_items_by_characteristic(customer, items)  
    filtered_items = filter_items_by_vetted_int(customer, filtered_items)  
    filtered_items = filter_items_by_vetted_item(customer, filtered_items)  
    return (filtered_items)
```

Listing B.7: Filter items by characteristic

```
def filter_items_by_characteristic(customer: Customer, items):  
    removed_items = []  
    for i in items:  
        item_filter = remove_item_by_characteristic(customer, i)  
        if item_filter :  
            removed_items.append(item_filter)  
    filtered_items = remove_filtered_items(removed_items, items)  
    return filtered_items
```

Listing B.8: Filter items by vetted interest

```
def filter_items_by_vetted_int(customer:Customer, items):
    filtered_items=items
    vetted_int = get_vetted_interests(customer)
    for j in items[:]:
        filtered_items = remove_vetted_item_by_int(j, vetted_int, filtered_items)
    return (filtered_items)
```

Listing B.9: Filter items by vetted item

```
def filter_items_by_vetted_item(customer:Customer, items):
    filtered_items=items
    vetted_item = get_vetted_items(customer)
    for j in items[:]:
        filtered_items = remove_vetted_item(j, vetted_item, filtered_items)
    return (filtered_items)
```

Listing B.10: Measure distance between filtered items and user profile

```
def make_customer_recommendation(customer, items, n_rec:int):
    recommendations = []
    profile = customer.drop(["ID", "name"], axis=1)
    profile = profile.values
    X = items.drop(["ID", "name", 'User_Preference'], axis=1)
    X = X.values
    distances = np.linalg.norm(X - profile, axis=1)
    nearest_neighbor_ids = distances.argsort()[:n_rec]
    for n in nearest_neighbor_ids:
        dicts_record = {'ID': items.iloc[n]['ID'], 'name' : items.iloc[n]['name'],
            'user_preference': items.iloc[n]['User_Preference']}
        recommendations.append(dicts_record)
    recommendations = sorted(recommendations, key=lambda d:
    d['user_preference'], reverse=True)
    return recommendations
```

Listing B.11: Return top recommended items

```
def return_recommendations(recommendations, items):
    recommendation_array=[]
    for i in recommendations:
        recommendation_array.append(find_by_id(i['ID'], items))
    return recommendation_array
```

Listing B.12: Add recommended items to session

```
def add_recommendations(session:Session, recommendation_array, customer):
```

APPENDIX B. RECOMMENDATION MODULE

```
for i in recommendation_array:
    weight=find_weight_in_customer(customer, i)
    session.add_item(i, False, weight, recommendation_array.index(i),
        recommendation_array.index(i))
return(session)
```

Appendix C

Dynamization module

Listing C.1: Analyze latest user feedback to modify interest weights

```
def analyze_latest_sessions(session_array, customer, interest_categories,
number_up, number_down, date):
    all_up_categories=[]
    all_down_categories=[]
    for i in session_array:
        session_date=get_date(i.timestamp)
        if(session_date > date):
            up_categories, down_categories = analyze_session_questions(i,
customer, interest_categories,
number_up, number_down)
            all_up_categories, all_down_categories =
append_all_up_down(all_up_categories, all_down_categories,
up_categories, down_categories)
            remove_from_array(all_up_categories, [])
            remove_from_array(all_down_categories, [])
    up=count_instances(all_up_categories)
    down=count_instances(all_down_categories)
    apply_modifications(up, customer.interests, "up")
    apply_modifications(down, customer.interests, "down")
return customer
```

Listing C.2: Analyze recommended items in each session

```
def analyze_question_items(item_array, customer, interest_categories,
number_up, number_down):
    all_up_categories=[]
    all_down_categories=[]
    first_recommended, first_final = get_first_recommended_position(item_array)
    for i in item_array:
        if (i['recommended_position']-i['final_position']) >= number_up:
```

```
up_categories, down_categories =
get_modified_categories(first_recommended, i['item'], customer,
interest_categories, 'up')
up_categories= add_position_difference(up_categories,
(i['recommended_position']-i['final_position']))
down_categories= add_position_difference(down_categories,
(i['recommended_position']-i['final_position']))
all_up_categories.extend(up_categories)
all_down_categories.extend(down_categories)
elif (i['recommended_position']-i['final_position']) <= number_down:
up_categories, down_categories=
get_modified_categories(first_final, i['item'], customer,
interest_categories, 'down')
up_categories= add_position_difference(up_categories,
(i['recommended_position']-i['final_position']))
down_categories= add_position_difference(down_categories,
(i['recommended_position']-i['final_position']))
all_up_categories.extend(up_categories)
all_down_categories.extend(down_categories)
return all_up_categories, all_down_categories
```

Listing C.3: Save interest categories that will be modifies

```
def compare_categories(item_dataset, type):
item_dataset = item_dataset.drop(["ID", "name"], axis=1)
up_categories=[]
down_categories=[]
for i in item_dataset.columns:
recommended = item_dataset.loc[0,i]
current_item = item_dataset.loc[1,i]
if(recommended == 0 and current_item == 1):
if type == 'up':
up_categories.append([i])
elif type == 'down':
down_categories.append([i])
elif(recommended == 1 and current_item == 0):
if type == 'up':
down_categories.append([i])
elif type == 'down':
up_categories.append([i])
return up_categories, down_categories
```

Listing C.4: Count instances of interest categories

```
def count_instances(array):
```

```
new_array = []
final_arr=[]
mean_arr = []
for i in array:
    new_array.append(i[0])
for j in new_array:
    for k in array:
        if k[0] == j:
            mean_arr.append(k[1])
    add = [j, statistics.mean(mean_arr), new_array.count(j)]
    if add not in final_arr:
        final_arr.append(add)
    mean_arr=[]
return(final_arr)
```

Listing C.5: Dynamization algorithm

```
def dynamization_algorithm(interest, mean_pos, frequency, type):
    initial_value = interest["explicit_weight"]
    if type == "up":
        res = initial_value + (initial_value * (mean_pos+frequency)/100)
    elif type == "down":
        res = initial_value - (initial_value * (mean_pos+frequency)/100)
    interest["implicit_weight"] = res
    return interest
```

Appendix D

Article database

D.1 Excel table

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
1	CP09	Personalized Recommendation on Dynamic Content Using Predictive Bilinear Models	2009	Wei Chu; Seung-Taek Park	Conference article	Association for Computing Machinery (ACM)	WWW '09: Proceedings of the 18th international conference on World wide web	Recommender systems; personalized systems; Dynamic features; User profile; Content profile; Evaluation	<p>In Web-based services of dynamic content (such as news articles), recommender systems face the difficulty of timely identifying new items of high-quality and providing recommendations for new users.</p> <p>We propose a feature-based machine learning approach to personalized recommendation that is capable of handling the cold-start issue effectively.</p> <p>We maintain profiles of content of interest, in which temporal characteristics of the content, e.g. popularity and freshness, are updated in real-time manner.</p> <p>We also maintain profiles of users including demographic information and a summary of user activities within Yahoo! properties.</p> <p>Based on all features in user and content profiles, we develop predictive bilinear regression models to provide accurate personalized recommendations of new items for both existing and new users.</p> <p>This approach results in an offline model with light computational overhead compared with other recommender systems that require online re-training.</p>
2	TM07	A Survey of Explanations in Recommender Systems	2007	Nava Tintarev; Judith Masthoff	Conference article	IEEE	2007 IEEE 23rd International Conference on Data Engineering Workshop	Recommender systems; personalized systems; Explanation facility; Usability	<p>This paper provides a comprehensive review of explanations in recommender systems.</p> <p>We highlight seven possible advantages of an explanation facility, and describe how existing measures can be used to evaluate the quality of explanations.</p> <p>Since explanations are not independent of the recommendation process, we consider how the ways recommendations are presented may affect explanations.</p> <p>Next, we look at different ways of interacting with explanations.</p> <p>The paper is illustrated with examples of explanations throughout, where possible from existing applications.</p>
3	IFO15	Recommendation systems: Principles, methods and evaluation	2015	F.O. Isinkaye; Y.O. Folajimi; B.A. Ojokoh	Journal article	Elsevier	Egyptian informatics journal	Recommender systems; Collaborative filtering; Content-based filtering; Hybrid filtering; Evaluation	<p>On the Internet, where the number of choices is overwhelming, there is need to filter, prioritize and efficiently deliver relevant information in order to alleviate the problem of information overload, which has created a potential problem to many Internet users.</p> <p>Recommender systems solve this problem by searching through large volume of dynamically generated information to provide users with personalized content and services.</p> <p>This paper explores the different characteristics and potentials of different prediction techniques in recommendation systems in order to serve as a compass for research and practice in the field of recommendation systems.</p>
4	BKN07	The Adaptive Web Methods and Strategies of Web Personalization	2007	Peter Brusilovski; Alfred Kobza; Wolfgang Nejdl	Book	Springer	Springer Link	Recommender systems; Adaptive systems; Survey; Collaborative filtering; Content-based filtering; Hybrid filtering; Data	<p>In the first few years after its inception, the Web was the same for everyone.</p> <p>Web sites presented the same information and the same links to all visitors, regardless of their goals and prior knowledge.</p> <p>A query to a Web search engine or catalog produced the same result for all users, irrespective of their underlying interests and information needs.</p> <p>With the growth of the available information on the Web, the diversity of its users and the complexity of Web applications, researchers started to question this "one-size-fits-all" approach. ...</p>
5	BOHG13	Recommender systems survey	2013	J. Bobadilla; F. Ortega; A. Hernandez; A. Gutierrez	Journal article	Elsevier	Knowledge-based systems	Recommender systems; Survey; Collaborative filtering; Similarity measures; Evaluation; Hybrid filtering; Cold-start	<p>Recommender systems have developed in parallel with the web. They were initially based on demographic, content-based and collaborative filtering. Currently, these systems are incorporating social information.</p> <p>In the future, they will use implicit, local and personal information from the Internet of things.</p> <p>This article provides an overview of recommender systems as well as collaborative filtering methods and algorithms; it also explains their evolution, provides an original classification for these systems, identifies areas of future implementation and develops certain areas selected for past, present or future importance.</p>
6	ZYST19	Deep Learning Based Recommender System: A Survey and New Perspectives	2019	S Zhang; L Yao; A Sun; Y Tay	Journal article	Association for Computing Machinery (ACM)	ACM Computing Surveys (CSUR)	Recommender systems; Survey; Deep learning; Neural networks	<p>With the growing volume of online information, recommender systems have been an effective strategy to overcome information overload.</p> <p>The utility of recommender systems cannot be overstated, given their widespread adoption in many web applications, along with their potential impact to ameliorate many problems related to over-choice.</p> <p>In recent years, deep learning has garnered considerable interest in many research fields such as computer vision and natural language processing, owing not only to stellar performance but also to the attractive property of learning feature representations from scratch.</p> <p>The influence of deep learning is also pervasive, recently demonstrating its effectiveness when applied to information retrieval and recommender systems research.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
7	AT05	Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions	2005	G. Adomavicius; A. Tuzhilin	Journal article	IEEE	IEEE Transactions on Knowledge and Data Engineering	Recommender Systems; Collaborative filtering; Content-based filtering; Hybrid filtering; rating estimation methods, extensions	<p>This paper presents an overview of the field of recommender systems and describes the current generation of recommendation methods that are usually classified into the following three main categories: content-based, collaborative, and hybrid recommendation approaches.</p> <p>This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities and make recommender systems applicable to an even broader range of applications.</p> <p>These extensions include, among others, an improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multicriteria ratings, and a provision of more flexible and less intrusive types of recommendations.</p>
8	YDH21	GFE: General Knowledge Enhanced Framework for Explainable Sequential Recommendation	2021	Zuoxi Yang; Shoubin Dong; Jinlong Hu	Journal article	Elsevier	Knowledge-Based Systems	Recommendation systems; Knowledge graph; Deep learning; Explanation facility; Sequential recommendation; Hierarchical self-	<p>It is vital for sequential recommendation to provide accurate and explainable results for user, which can help them make better decisions.</p> <p>In this paper, we develop a General Knowledge Enhanced Framework for Explainable Sequential Recommendation (GFE) to capture user's fine-grained preferences and dynamic preferences evolution.</p> <p>Specifically, the fine-grained preferences are modeled as intrinsic interests and external potential interests, which can be captured by sequential-aware interest and knowledge-aware interest modules respectively.</p> <p>Moreover, the high-order paths between each user-item pair are generated with the help of the knowledge graph, which contain abundant high-order semantic relevance among entities.</p> <p>To make better use of this character, we propose a hierarchical self-attention mechanism to</p>
9	YCC+15	Dynamic User Modeling in Social Media Systems	2015	Hongzhi Yin; Bin Cui; Ling Chen; Zhiting Hu; Xiaofang Zhou	Journal article	Association for Computing Machinery (ACM)	ACM Transactions on Information Systems (TOIS)	Profile dynamics; Algorithms, Design, Experimentation, Performance	<p>Social media provides valuable resources to analyze user behaviors and capture user preferences. This article focuses on analyzing user behaviors in social media systems and designing a latent class statistical mixture model, named temporal context-aware mixture model (TCAM), to account for the intentions and preferences behind user behaviors.</p> <p>Based on the observation that the behaviors of a user in social media systems are generally influenced by intrinsic interest as well as the temporal context (e.g., the public's attention at that time), TCAM simultaneously models the topics related to users' intrinsic interests and the topics related to temporal context and then combines the influences from the two factors to model user behaviors in a unified way.</p> <p>Considering that users' interests are not always stable and may change over time, we extend TCAM to a dynamic temporal context-aware mixture model (DTCAM) to capture users' changing interests.</p>
10	THHK21	User-Centric Path Reasoning towards Explainable Recommendation	2021	Changyou Tai; Liangying Huang; Chienkun Huang; Lunwei Ku	Conference article	Association for Computing Machinery (ACM)	SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval	Recommendation systems; Knowledge graph; Deep learning; Explanation facility	<p>There has been significant progress in the utilization of heterogeneous knowledge graphs (KG) as auxiliary information in recommendation systems. Reasoning over KG paths sheds light on the user's decision-making process.</p> <p>Previous methods focus on formulating this process as a multi-hop reasoning problem.</p> <p>However, without some form of guidance in the reasoning process, such a huge search space results in poor accuracy and little explanation diversity. In this paper, we propose UCPR, a user-centric path reasoning network that constantly guides the search from the aspect of user demand and enables explainable recommendations. In this network, a multi-view structure leverages not only local sequence reasoning information but also a panoramic view of the user's demand portfolio while inferring subsequent user decision-making steps.</p> <p>Experiments on five real-world benchmarks show UCPR is significantly more accurate than state-of-the-art methods. Besides, we show that the proposed model successfully identifies</p>
11	CLS+21	Temporal Meta-path Guided Explainable Recommendation	2021	Hongxu Chen; Yicong Li; Xiangguo Sun; Guandong Xu; Hongzhi Yin	Conference article	Association for Computing Machinery (ACM)	WSDM '21: Proceedings of the 14th ACM International Conference on Web Search and Data Mining	Recommendation systems; Deep learning; Explanation facility; Profile Dynamics; Data mining	<p>Recent advances in path-based explainable recommendation systems have attracted increasing attention thanks to the rich information provided by knowledge graphs.</p> <p>Most existing explainable recommendation only utilizes static knowledge graph and ignores the dynamic user-item evolutions, leading to less convincing and inaccurate explanations.</p> <p>Although there are some works that realize that modelling user's temporal sequential behaviour could boost the performance and explainability of the recommender systems, most of them either only focus on modelling user's sequential interactions within a path or independently and separately of the recommendation mechanism. In this paper, we propose a novel Temporal Meta-path Guided Explainable Recommendation (TMER), which utilizes well-designed item-item path modelling between consecutive items with attention mechanisms to sequentially model dynamic user-item evolutions on dynamic knowledge graph for explainable recommendations.</p>
12	WHW+22	A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation	2022	Le Wu; Xiangnan He; Xiang Wang; Kun Zhang; Meng Wang	Journal article	IEEE	IEEE Transactions on Knowledge and Data Engineering	Recommender systems; Survey; Deep learning; Neural Networks; Neural Recommendation Models	<p>Influenced by the great success of deep learning in computer vision and language understanding, research in recommendation has shifted to inventing new recommender models based on neural networks.</p> <p>In recent years, we have witnessed significant progress in developing neural recommender models, which generalize and surpass traditional recommender models owing to the strong representation power of neural networks.</p> <p>In this survey paper, we conduct a systematic review on neural recommender models, aiming to summarize this field to facilitate researchers and practitioners working on recommender systems.</p> <p>Specifically, based on the data usage during recommendation modeling, we divide the work into collaborative filtering and information-rich recommendation:</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
13	TNM18	Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning	2018	JohnK. Tarus; Zhendong Niu; Ghulam Mustafa	Journal article	Springer	Artificial intelligence review	Recommender systems; Knowledge-based RS; E-learning; Ontology	<p>Recommender systems in e-learning domain play an important role in assisting the learners to find useful and relevant learning materials that meet their learning needs.</p> <p>Personalized intelligent agents and recommender systems have been widely accepted as solutions towards overcoming information retrieval challenges by learners arising from information overload.</p> <p>Use of ontology for knowledge representation in knowledge-based recommender systems for e-learning has become an interesting research area. In knowledge-based recommendation for e-learning resources, ontology is used to represent knowledge about the learner and learning resources.</p> <p>Although a number of review studies have been carried out in the area of recommender systems, there are still gaps and deficiencies in the comprehensive literature review and survey in the specific area of ontology-based recommendation for e-learning. In this paper,</p>
14	LMNS19	Semantics in Adaptive and personalized Systems: Methods, Tools and Applications	2019	Pasquale Lops; Cataldo Musto; Fedelucio Narducci; Giovanni Semeraro;	Book	Springer	Springer Link	personalized systems; Adaptive systems	<p>Broadens your understanding of semantics-aware content representation techniques and reviews exhaustively the key recent research in the area</p> <p>Explains how to effectively exploit techniques for semantics-aware content representation to improve the effectiveness of adaptive and personalized systems</p> <p>Contains supplementary material on the available tools and APIs to develop your own semantics-aware adaptive and personalized system</p> <p>Discusses new trends and challenges including Serendipitous Recommendations, Conversational Recommender Systems, and Explanation Algorithms</p>
15	SA19	Knowledge Based Recommender System for Academia Using Machine Learning: A Case Study on Higher Education Landscape	2019	Huma Samin; Tayyaba Azim	Journal article	IEEE	IEEE Access	Knowledge-based recommender system; knowledge management application; latent Dirichlet allocation; machine learning;	<p>Allocation of courses and research students based on faculty's subject specialization and area of interest has always remained a challenging task for university administration due to the presence of academics' cross-domain interests, stale faculty resumes at university portals and changing the skill set demands from the industry.</p> <p>Collaborative filtering and content-based recommender systems have already been in use by the industry for recommending things, such as movies, news, restaurants, and shopping items to the users, and however, no one has utilized these off-the-shelf models for enhancing the student experience and improving the quality of higher education in academia.</p> <p>This paper presents a case study showcasing the use of probabilistic topic models for generating recommendations to users in academia through appropriate course allocation and supervisor assignment.</p> <p>The proposed system coined as ScholarLite harnesses the power of machine learning to</p>
16	FFJZ06	An integrated environment for the development of knowledge-based recommender applications	2006	Alexander Felfernig; Gerhard Friedrich; Dietmar Jannach M.S.; Markus	Journal article	Taylor & Francis	International Journal of Electronic Commerce	End-user programming; knowledge acquisition; knowledge-based recommenders; model-based diagnosis;	<p>The complexity of the product assortments offered by on-line selling platforms makes selection a challenging task.</p> <p>Customers differ in respect to expertise and product knowledge, but intelligent recommender systems offer personalized dialogues that support the product-selection process.</p> <p>This paper describes CWAdvisor, a domain-independent, knowledge-based recommender environment that provides users with consistently appropriate solutions, identifies additional selling opportunities, and explains solutions.</p> <p>The discussion uses examples from several application domains to show how model-based diagnosis, personalization, and intuitive knowledge-acquisition techniques support customer-oriented sales dialogues. Experience obtained in industrial projects is reported, and successfully deployed recommender applications are evaluated.</p>
17	CC12	Semantically-Enhanced Recommenders	2012	Victor Codina; Luigi Ceccaroni	Conference article	IOS Press	Artificial Intelligence Research and Development: Proceedings of the 15th International Conference of the Catalan	Knowledge-based recommender systems	<p>Recommendation models based on collaborative filtering have good prediction accuracy; however, they have very poor performance in cold-start scenarios, when no or few rating data are associated to certain items or users. An approach for dealing with cold-start problem is to build hybrid models that include content-based filtering.</p> <p>This paper focuses on knowledge-based hybrids that exploit semantic relations among item attributes to enhance traditional models. We investigate novel methods for exploiting this semantics during the recommendation process, in domains where explicit semantic relations among attributes in the form of ontologies are not available. We carry out an extensive performance comparison in which the proposed semantically-enhanced models are evaluated in normal and cold-start scenarios.</p> <p>The experimental results with a publicly available dataset demonstrate superior better performance of the proposed semantically-enhanced models, above all in cold-start scenarios, with respect to a state-of-the-art collaborative-filtering model based on matrix</p>
18	CMVGRG+15	RecomMetz: A context-aware knowledge-based mobile recommender system for movie showtimes	2015	Luis Omar Colombo-Mendoza; Rafael Valencia-García; Alejandro Rodríguez-	Journal article	Elsevier	Expert Systems with Applications	Knowledge-based recommender systems; Context-aware systems; Semantic Web; Ontology reasoning	<p>Recommender systems are used to provide filtered information from a large amount of elements. They provide personalized recommendations on products or services to users.</p> <p>The recommendations are intended to provide interesting elements to users. Recommender systems can be developed using different techniques and algorithms where the selection of these techniques depends on the area in which they will be applied.</p> <p>This paper proposes a recommender system in the leisure domain, specifically in the movie showtimes domain. The system proposed is called RecomMetz, and it is a context-aware mobile recommender system based on Semantic Web technologies.</p> <p>In detail, a domain ontology primarily serving a semantic similarity metric adjusted to the concept of "packages of single items" was developed in this research. In addition, location, crowd and time were considered as three different kinds of contextual information in RecomMetz.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
19	HF15	Dynamic user profiles for web personalisation	2015	Ahmad Hawalah; Maria Fasli	Journal article	Elsevier	Expert Systems with Applications	Profile dynamics; Personalization; Content-Based Recommendation; Modelling user behaviour; Web personalisation; Ontology;	<p>Web personalisation systems are used to enhance the user experience by providing tailor-made services based on the user's interests and preferences which are typically stored in user profiles.</p> <p>For such systems to remain effective, the profiles need to be able to adapt and reflect the users' changing behaviour. In this paper, we introduce a set of methods designed to capture and track user interests and maintain dynamic user profiles within a personalisation system. User interests are represented as ontological concepts which are constructed by mapping web pages visited by a user to a reference ontology and are subsequently used to learn short-term and long-term interests.</p> <p>A multi-agent system facilitates and coordinates the capture, storage, management and adaptation of user interests. We propose a search system that utilises our dynamic user profile to provide a personalized search experience. We present a series of experiments that show how our system can effectively model a dynamic user profile and is capable of learning</p>
20	RDS+21	An intelligent location recommender system utilising multi-agent induced cognitive behavioural model	2021	Logesh Ravi, Malathi Devarajan, Vijayakumar V, Arun Kumar Sangai ah, Lipo Wang,	Journal article	Taylor & Francis	Enterprise Information Systems	Profile Dynamics, Collaborative Filtering; computational Intelligence; multi-Agent Systems; recommender	<p>In recent times, location recommendation has received significant attention from the researchers due to emerging utilisation of Location Based Social Networks in the prediction process.</p> <p>In this paper, we present a new multi-agent based framework to generate better-personalized location recommendations. We address the personalisation problem through the dynamic user profile that incorporates the user's long-term and short-term cognitive behaviour.</p> <p>The better adaptation of user cognitive behaviour enhances the prediction process and improves overall user experience with better recommendations.</p> <p>A detailed user study is conducted to reveal the improved performance of proposed approach through enhanced recommendations in comparison with other approaches.</p>
21	TDW X21	Bidirectional sensing of user preferences and application changes for dynamic mobile app recommendations	2021	Zhiying Tu; Binghua Duan; Zhongjie Wang; Xiaofei Xu	Journal article	Springer	Neural Computing and Applications	Profile Dynamics; Mobile app recommendation; Personal interest evolution network; User preference	<p>Recent years have witnessed the rapid adoption of mobile devices and significant growth in the use of mobile apps.</p> <p>However, the large number of mobile apps makes it difficult for users to determine which ones are of interest. Current app recommendation methods focus on the similarities between user interest and mobile app functionality at a single moment, but they do not consider how these change over time and do not address the underlying mutual effect of such changes.</p> <p>This paper describes a bidirectional sensing method for user interest and app changes that enables precise mobile app recommendations. The proposed method applies continuous bidirectional sensing to both user interest and the functional changes to apps and obtains the evolution trajectory of both. Then, a personal interest evolution network is used to learn the relationship between the two trajectories.</p> <p>Finally, user interest in an app is predicted, and the top-N strategy is used to recommend</p>
22	RHS+13	SMARTMUSEUM: A mobile recommender system for the Web of Data	2013	Tuukka Ruotsalo; Krister Haav; Antony Stoyanov; Sylvain Roche; Elena Fani;	Journal article	Elsevier	Journal of Web Semantics	Knowledge-based recommenders; Semantic Web; Information retrieval; Recommender systems; Mobile systems; Cultural	<p>Semantic and context knowledge has been envisioned as an appropriate solution for addressing the content heterogeneity and information overload in mobile Web information access, but few have explored their full potential in mobile scenarios, where information objects refer to their physical counterparts, and retrieval is context-aware and personalized for users.</p> <p>We present SMARTMUSEUM, a mobile ubiquitous recommender system for the Web of Data, and its application to information needs of tourists in context-aware on-site access to cultural heritage. The SMARTMUSEUM system utilizes Semantic Web languages as the form of data representation.</p> <p>Ontologies are used to bridge the semantic gap between heterogeneous content descriptions, sensor inputs, and user profiles. The system makes use of an information retrieval framework wherein context data and search result clustering are used in recommendation of suitable content for mobile users.</p>
23	Puk14	LORecommNet: An Ontology-Based Representation of Learning Object Recommendation	2014	Noppamas Pukkhem	Book Chapter	Springer	Recent Advances in Information and Communication Technology	Ontology; Semantic Web	<p>One of the most problems facing learners in e-learning system is to find the most suitable course materials or learning objects for their personalized learning space.</p> <p>The main focus of this paper is to extend our previous rule-based representation recommendation system [1] by applying an ontology-based approach for creating a semantic learning object recommendation named "LORecommNet".</p> <p>The "LORecommNet" ontology represents the knowledge about learning objects, learner model, semantic mapping rules and their relationship are proposed. In the proposed framework, we demonstrated how the "LORecommNet" can be used to enable machines to interpret and process learning object in recommendation system. We also explain how ontological representations play a role in mapping learner to personalized learning object.</p> <p>The structure of "LORecommNet" extends the semantic web technology, which the representation of each based on an OWL ontology and then on the inference layer, based on</p>
24	AION 15	A reference ontology for profiling scholar's background knowledge in recommender systems	2015	Bahram Amini; Roiana Ibrahim; Mohd Othman; Mohammad Nematbakhsh	Journal article	Elsevier	Expert Systems with Applications	Ontology; Profiling; Semantic Web; DBpedia; Scholars; Recommender system	<p>The profiling of background knowledge is essential in scholar's recommender systems. Existing ontology-based profiling approaches employ a pre-built reference ontology as a backbone structure for representing the scholar's preferences.</p> <p>However, such singular reference ontologies lack sufficient ontological concepts and are unable to represent the hierarchical structure of scholars' knowledge. They rather encompass general-purpose topics of the domain and are inaccurate in representing the scholars' knowledge.</p> <p>This paper proposes a method for integrating of multiple domain taxonomies to build a reference ontology, and exploits this reference ontology for profiling scholars' background knowledge. In our approach, various topics of Computer Science domain from Web taxonomies are selected, transformed by DBpedia, and merged to construct a reference ontology.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
25	KSYF18	AK Tourism: A Property Graph Ontology-based Tourism Recommender System	2018	Keng-Seong She; Su-Cheng Haw; Yu-Gen Loh; Fang-Fang Chua	Conference article	UNIV UTARA MALAYSIA PRESS	9th Knowledge Management International Conference (KMICe)	ontology; recommender system; user profiling; tourism; point of interest.	<p>Recommender system (RS) is used to overcome the problem of information overload over the World Wide Web. Most of the time, users will be returned with a huge number of links when they do a Point of Interest (POI) search for interesting places to visit via the search engine.</p> <p>RS overcomes the problem by analyzing automatically all the possible items information integrates with personalized user profiling (user preferences), and user rating. Specifically, ontology-based recommender system adopts new and alternative trends in recent years.</p> <p>Ontologies define rules to structure data, including interrelations between entities in the database. As such, it offers greater semantic relations within a particular domain. In this paper, we propose: (i) ontology representation, (ii) sample query in Cypher, and (iii) AK Tourism to demonstrate the recommender system in Malaysia tourism domain.</p>
26	MGA+15	Using Process Ontologies to Contextualize Recommender Systems in Engineering Projects for Knowledge	2015	Mahsa Mehrpour; Jon Atle Gulla; Dirk Ahlers; Kjetil Kristensen; Ole Ivar Sivertsen; Sorous	Conference article	Academic Conferences and Publishing International	16th European Conference on Knowledge Management (EKCM)	Semantic Web; Ontology; Profile dynamics; Work situation; Recommender Systems; Semantic recommendation; Process ontology;	<p>Knowledge and information are highly important resources in today's knowledge economy and vital in achieving organization's goals. Particularly in engineering projects, users' tasks are highly dependent on accessing, using, and reusing these resources and users already spend a lot of time searching for relevant knowledge.</p> <p>As the total volume of documents across different sources and repositories increases, users face additional overhead related to search and retrieval. Knowledge workers across multiple disciplines experience fierce competition and a persistent pressure to deliver value-added contributions in a competitive global business environment with complex, multidisciplinary problems.</p> <p>Simple search engines are often not sufficient since they are not designed to retrieve those relevant documents that match the user's current work situation and information need. Therefore, the need for a semantics-based solution has been identified. This paper describes the early stages of a PhD project that proposes a tailored recommender system for improving</p>
27	dPCS13	A Hierarchical Architecture for Ontology-Based Recommender Systems	2013	Fábio Augusto Procópio de Paiva; José Alfredo Ferreira Costa; Cláudio Rodrigo	Conference article	IEEE	2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence	Ontology; Semantic Web; user profile; recommendation algorithm	<p>There is no doubt that the World Wide Web has made easier the task of searching for information on the Internet. The amount of information obtained (some of them irrelevant ones) increases day after day and creates opportunities for a new breed of systems named "Recommender Systems".</p> <p>These systems have emerged as one successful approach to tackle the problem of information overload. Traditional recommender systems suggest research items using well-known text mining techniques, however they fail when there are no identical keywords to match searches.</p> <p>In order to overcome this and other limitations, several studies have been made in order to verify the benefits of ontology-based approaches to create what is known as ontology-based recommender systems.</p> <p>This paper analyzes several ontology-based recommender systems and discusses some</p>
28	OLEK C18	Ontology-based Recommender System in Higher Education	2018	Obeid, Charbel and Lahoud, Inaya and El Khoury, Hicham and Champin, Pierre-Antoine	Conference article	Association for Computing Machinery (ACM)	WWW '18: Companion Proceedings of the Web Conference 2018	Ontology; Education; Recommender system	<p>Academic advising is limited in its ability to assist students in identifying academic pathways. Selecting a major and a university is a challenging process rife with anxiety. Students at high school are not sure how to match their interests with their working future or major.</p> <p>Therefore, high school students need guidance and support. Moreover, students need to filter, prioritize and efficiently get appropriate information from the web in order to solve the problem of information overload. This paper represents an approach for developing ontology-based recommender system improved with machine learning techniques to orient students in higher education.</p> <p>The proposed recommender system is an assessment tool for students' vocational strengths and weaknesses, interests and capabilities. The main objective of our ontology-based recommender system is to identify the student requirements, interests, preferences and capabilities to recommend the appropriate major and university for each one.</p>
29	CKL15	Ontology-based library recommender system using MapReduce	2015	Lun-Chi Chen, Ping-Jen Kuo & I-En Liao	Journal article	Springer	Cluster Computing	Ontology; MapReduce; Collaborative filtering; Recommender system	<p>Recommender systems have been proven useful in numerous contemporary applications and helping users effectively identify items of interest within massive and potentially overwhelming collections.</p> <p>Among the recommender system techniques, the collaborative filtering mechanism is the most successful; it leverages the similar tastes of similar users, which can serve as references for recommendation. However, a major weakness for the collaborative filtering mechanism is its performance in computing the pairwise similarity of users.</p> <p>Thus, the MapReduce framework was examined as a potential means to address this performance problem. This paper details the development and employment of the MapReduce framework, examining whether it improves the performance of a personal ontology based recommender system in a digital library.</p> <p>The results of this extensive performance study show that the proposed algorithm can scale</p>
30	DSR O20	Weighted aspect-based opinion mining using deep learning for recommender system	2020	Aminu Da'u; Naomie Salima; Idris Rabiua; Akram Osmana	Journal article	Elsevier	Expert Systems with Applications	Aspect-based opinion mining; Convolutional neural network; Deep learning; Collaborative filtering; Recommender system;	<p>The main goal of Aspect-Based Opinion Mining is to extract product's aspects and the associated user opinions from the user text review. Although this serves as vital source information for enhancing rating prediction performance, few studies have attempted to fully utilize it for better accuracy of recommendation systems.</p> <p>Most of these studies typically assign equal weights to all aspects in the opinion mining process, however, in practices; users tend to give different priority on different aspects of the product when reaching overall ratings. In addition, most of the existing methods typically rely on handcrafted, rule-based or double propagation methods in the opinion mining process which are known to be time-consuming and often inclined to errors.</p> <p>This could affect the reliability and performance of the recommender systems (RS). Therefore, in this paper, we propose a weighted Aspect-based Opinion mining using Deep learning method for Recommender system (AODR) that can extract product's aspects and the underlying weighted user opinions from the review text using a deep learning method and</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
31	SCP12	Effective Tag Recommendation System Based on Topic Ontology Using Wikipedia and WordNet	2012	V. Subramaniyaswamy; S. Chentur Pandian	Journal article	Wiley Online Library	International journal of intelligent systems	Ontology; User profiling;	In this paper, we proposed a novel approach based on topic ontology for tag recommendation. The proposed approach intelligently generates tag suggestions to blogs. In this approach, we construct topic ontology through enriching the set of categories in existing small ontology called as Open Directory Project. To construct topic ontology, a set of topics and their associated semantic relationships is identified automatically from the corpus-based external knowledge resources such as Wikipedia and WordNet. The construction relies on two folds such as concept acquisition and semantic relation extraction. In the first fold, a topic-mapping algorithm is developed to acquire the concepts from the semantic of Wikipedia. A semantic similarity-clustering algorithm is used to compute the semantic similarity measure to group the set of similar concepts. The second is the semantic relation extraction algorithm, which derives associated semantic
32	LJZ08	Ontology-based user modeling for e-commerce system	2008	Weilong Liu; Fang Jin; Xin Zhang	Conference article	IEEE	2008 Third International Conference on Pervasive Computing and Applications	Ontology; User modeling; E-commerce	User modeling, often addressed as user profile and a simple way of capturing his/her preferences, plays key roles in marketing and personalized services in E-commerce. Due to its powerful knowledge representation and inference mechanism, ontology-based user modeling is emerging as a natural choice to represent knowledge about users in E-commerce context. This paper has presented ontology-based user modeling system, and relies on user ontology to provide customized services in the context of E-commerce. A generic framework of implicit and explicit ontology-based user modeling is presented and described using Semantic Web technology, i.e. OWL specification, which can improve the quality of returned information for user's queries.
33	XZZG10	Context-aware content filtering and presentation for pervasive and mobile information systems	2010	Kaijian Xu; Manli Zhu; Daqing Zhang; Tao Gu	Conference article	ICST	1st International ICST Conference on Ambient Media and Systems	context awareness, ambient intelligence, content integration, filtering, presentation, adaptivity, OLAP, preference mining; User	What constitutes relevant information to an individual may vary widely under different contexts. However, previous work on pervasive information systems has mostly focused on context-aware delivery of application-specific information. Such systems are only able to operate within narrow application domains and cannot be generalized to handle other heterogeneous types of information. To fill this gap, we propose a context-aware system for information integration that can handle arbitrary information types and determine their relevance to the user's current context. In contrast to existing model-based approaches to context reasoning, we log user interaction and perform usage mining using OLAP to discover context-dependent preferences for different information types. This allows us to build a more generic and adaptive system that automatically selects the most relevant content and presents it to the user in a succinct manner that supports ease of
34	WC08	Using ontology network analysis for research document recommendation	2008	Sung-Shun Weng; Hui-Ling Chang	Journal article	Elsevier	Expert Systems with Applications	Ontology; Ontology network analysis; Spreading activation model; Recommendation system	Following tremendous advancement in information technology, the speed of information development has become increasingly fast-paced. Yet the overabundance of information has forced users to spend more time and resources in searching for information relevant to their needs. Today, recommendation systems already exist that provide services like filtering, customization, and others to assist users in searching for the right information. This study proposes to use ontology and the spreading activation model for research paper recommendation, hoping that it can elevate the performance of the recommendation system and also improve the shortcomings of today's recommendation systems. This study utilizes ontology to construct user profiles and makes use of user profile ontology as the basis to reason about the interests of users. Furthermore, this study takes advantage of the spreading activation model to search for other influential users in the community network environment, making a study on their interests in order to provide recommendation
35	CFV+08	A Multi-Purpose Ontology-Based Approach for Personalized Content Filtering and Retrieval	2008	Iván Cantador; Miriam Fernández; David Vallet; Pablo Castells; Jérôme Picault; and	Book Chapter	Springer	Advances in Semantic Media Adaptation and Personalization	User Preference; Domain Ontology; User Interest; Decay Factor; Concept History	personalized multimedia access aims at enhancing the retrieval process by complementing explicit user requests with implicit user preferences. We propose and discuss the benefits of the introduction of ontologies for an enhanced representation of the relevant knowledge about the user, the context, and the domain of discourse, as a means to enable improvements in the retrieval process and the performance of adaptive capabilities. We develop our proposal by describing techniques in several areas that exemplify the exploitation of the richness and power of formal and explicit semantics descriptions, and the improvements therein. In addition, we discuss how those explicit semantics can be learnt automatically from the analysis of the content consumed by a user, determining which concepts appear to be significant for the user's interest representation.
36	FL07	Ontology-based user profiling	2007	Carsten Felden; Markus Linden	Conference article	Springer	International Conference on Business Information Systems	Profiling; Ontology; Recommender Systems; E-commerce	Profiles are the basis for individual communication, because they provide information about website users. Ontologies represent a possibility for modeling user profiles. The ontology development within the paper is based on a concept which shows firstly criteria of segmentation and secondly product programs of a retailer. A meta-ontology is built to enforce a mapping between the ontologies. Due to ontology-based recommendations it is obvious that they imply an additive character regarding conventional recommender systems. Therefore, the possibility arises to increase the turnover and to achieve customer satisfaction. But, the usage of ontology-based profiles is currently disputable relating to economic efficiency.

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
37	FEMR18	User Profiling Approaches, Modeling, and Personalization	2018	Marina Farid; Rania Elgohary; Ibrahim Moawad; Mohamed Roushdy	Conference article	ACM?	Proceedings of the 11th International Conference on Informatics & Systems (INFOS 2018)	User profiling, Ontology, Dynamic, Semantic similarity, Personalization	<p>The growth of the available information on the internet and the huge diversity of users give high priority to personalization.</p> <p>User profiling is the crucial issue for both information and service personalization. Building an automated user profile is the main challenge in developing an adaptive personalized application.</p> <p>Recent research works are involved in developing systems with personalizing user's interaction.</p> <p>This paper examines what information is needed to be modeled for presenting various user models, how this information is collected, how the user model is represented and maintained, and finally how the user model is exploited to deliver personalized services.</p> <p>The paper investigates the current studies on user profiling and modeling so far. A novel</p>
38	GXC M14	A Multi-agent Context-Based Personalized User Preference Profile Construction Approach	2014	Qian Gao; Su Mei Xi; Young Im Cho; Eric T. Matson	Conference article	Springer	Soft Computing in Advanced Robotics	Multi-Agent, Context, WordNet, Ontology.	<p>This paper proposes a Context-Based Personalized User Preference Profile Construction Approach to comprehensively track the user's local behaviors and user's web behavior of new inputted query.</p> <p>The traditional user profile construction may mainly consider the browsing behavior such as webpage click frequency and webpage click history, but lack consideration of local device context information. So, in this paper, we make use of the context information (interactive historical information and user information that related with the retrieval) which are stored and used in all of the smart devices, owned by the same user, to build and update the user preference profile.</p> <p>Furthermore, in order to avoid the limitation of different vector positions may be allocated to the synonyms of the same term, as well as the size of a document vector must be at least equal to the total number of the words used to write the document, we use the method of ontology-based representation based on WordNet, which uses WordNet to identify WordNet</p>
39	LYW K07	Dynamic Adaptation Strategies for Long-Term and Short-Term User Profile to Personalized Search	2007	Lin Li; Zhenglu Yang; Botao Wang; Masaru Kitsuregawa	Conference article	Springer	Advances in Data and Web Management	Search Engine; Search Result; Semantic Similarity; User Preference; User Interest;	<p>Recent studies on personalized search have shown that user preferences could be learned implicitly. As far as we know, these studies, however, neglect that user preferences are likely to change over time.</p> <p>This paper introduces an adaptive scheme to learn the changes of user preferences from click-history data, and a novel rank mechanism to bias the search results of each user. We propose independent models for long-term and short-term user preferences to compose our user profile.</p> <p>The proposed user profile contains a taxonomic hierarchy for the long-term model and a recently visited page-history buffer for the short-term model. Dynamic adaptation strategies are devised to capture the accumulation and degradation changes of user preferences, and adjust the content and the structure of the user profile to these changes.</p> <p>Experimental results demonstrate that our scheme is efficient to model the up-to-date user</p>
40	BWC +12	Modeling the impact of short- and long-term behavior on search personalization	2012	Paul N. Bennett, Ryan W. White, Wei Chu, Susan T. Dumais, Peter	Conference article	Association for Computing Machinery (ACM)	SIGIR '12: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval	Personalization; Web search.	<p>User behavior provides many cues to improve the relevance of search results through personalization. One aspect of user behavior that provides especially strong signals for delivering better relevance is an individual's history of queries and clicked documents.</p> <p>Previous studies have explored how short-term behavior or long-term behavior can be predictive of relevance. Ours is the first study to assess how short-term (session) behavior and long-term (historic) behavior interact, and how each may be used in isolation or in combination to optimally contribute to gains in relevance through search personalization.</p> <p>Our key findings include: historic behavior provides substantial benefits at the start of a search session; short-term session behavior contributes the majority of gains in an extended search session; and the combination of session and historic behavior out-performs using either alone.</p> <p>We also characterize how the relative contribution of each model changes throughout the</p>
41	DZD22	Improving Personalized Search with Dual-Feedback Network	2022	Chenlong Deng; Yujia Zhou; Zhicheng Dou	Conference article	Association for Computing Machinery (ACM)	Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining	Personalized search; Dual-feedback network; User intention	<p>Personalized search improves the quality of search results by modeling historical user behavior. In recent years, many methods based on deep learning have greatly improved the performance of personalized search. However, most of the existing methods only focus on modeling positive user behavior signals, which leads to incomplete user interest modeling.</p> <p>At the same time, the user's search behavior hides much explicit or implicit feedback information. For example, clicking and staying for a certain period represents implicit positive feedback, and skipping behavior represents implicit negative feedback.</p> <p>Intuitively, this information can be utilized to construct a more complete and accurate user profile. In this paper, we propose a dual-feedback modeling framework, which integrates multi-granular user feedback information to model the user's current search intention.</p> <p>Specifically, we propose a feedback extraction network to refine the dual-feedback representation in multiple stages. For enhancing the user's real-time search quality, we</p>
42	RJ14	An evolutionary clustering algorithm based on temporal features for dynamic recommender systems	2014	Chhavi Rana; Sanjay Kumar Jain	Journal article	Elsevier	Swarm and Evolutionary Computation	Evolutionary; Clustering; Algorithm; Recommender systems; Collaborative filtering; Data mining	<p>The use of internet and Web services is changing the way we use resources and communicate since the last decade. Although, this usage has made life easier in many respects still the problem of finding relevant information persists.</p> <p>A naive user faces the problem of information overload and continuous flow of new information makes the problem more complex. Furthermore, user's interests also keeps on changing with time. Several techniques deal with this problem and data mining is widely used among them.</p> <p>Recommender Systems (RSs) assist users in finding relevant information on the web and are mostly based on data mining algorithms. This paper addresses the problem of user requirements changing over a period of time in seeking information on web and how RSs deal with them.</p> <p>We propose a Dynamic Recommender system (DRS) based on evolutionary clustering</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
43	RJ14 pt 2	An extended evolutionary clustering algorithm for an adaptive recommender system	2014	Chhavi Rana; Sanjay Kumar Jain	Journal article	Springer	Social Network Analysis and Mining	Evolutionary; Clustering; Algorithm; Recommender systems; Collaborative filtering; Data mining	<p>The usage of internet and Web services has tremendously increased since the past decade. This usage has made life easier in many respects, especially in finding the required information with the help of search engines.</p> <p>However, the problem of finding relevant information still persists primarily due to an increasing number of choices presented by the search engines. As recommender systems (RSs) are evolving, these new generation RSs are adopted by many web services to build long-term relation with customers.</p> <p>RSs aid users in finding relevant information on the web. Several techniques deal with this problem and data mining is widely used among them.</p> <p>They target the information overload problem and also strive to present updated suggestions as new information arrives. The goal of presenting updated information to users is one of the foremost challenge in the area of RS research as user's interests also keeps on changing</p>
44	LPL+ 16	Collaborative Evolution for User Profiling in Recommender Systems	2016	Zhongqi Lu, Sinno Jialin Pan, Yong Li, Jie Jiang, Qiang Yang	Conference article	Hong Kong University of Science and Technology	Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)	Evolutionary; Recommender systems; Collaborative filtering; Data mining	<p>Accurate user profiling is important for an online recommender system to provide proper personalized recommendations to its users.</p> <p>In many realworld scenarios, the user's interests towards the items may change over time. Therefore, a dynamic and evolutionary user profile is needed. In this work, we come up with a novel evolutionary view of user's profile by proposing a Collaborative Evolution (CE) model, which learns the evolution of user's profiles through the sparse historical data in recommender systems and outputs the prospective user profile of the future.</p> <p>To verify the effectiveness of the proposed model, we conduct experiments on a real-world dataset, which is obtained from the online shopping website of Tencent Inc.- www.51buy.com and contains more than 1 million users' shopping records in a time span of more than 180 days.</p> <p>Experimental analyses demonstrate that our proposed CE model can be used to make better</p>
45	RJ15	A study of the dynamic features of recommender systems	2015	Chhavi Rana; Sanjay Kumar Jain	Journal article	Springer	Artificial Intelligence Review	Recommender systems · Collaborative filtering · Information overload · Dynamic · Temporal	<p>The extensive usage of internet is fundamentally changing the way we live and communicate. Consequently, the requirements of users while browsing internet are changing drastically.</p> <p>Recommender Systems (RSs) provide a technology that helps users in finding relevant contents on internet. Revolutionary innovations in the field of internet and their consequent effects on users have activated the research in the area of recommender systems.</p> <p>This paper presents issues related to the changing needs of user requirements as well as changes in the systems' contents. The RSs involving said issues are termed as Dynamic Recommender Systems (DRSs). The paper first defines the concept of DRS and explores the various parameters that contribute in developing a DRS.</p> <p>The paper also discusses the scope of contributions in this field and concludes citing in possible extensions that can improve the dynamic qualities of recommendation systems in future.</p>
46	EHNA 19	Queries-based Profile Evolution using Genetic Algorithm	2019	Boukri nat Nour El Houda; Benblidia Nadjia; Meziane Abdelkrim	Conference article	IEEE	2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)	User profile · Evolution · User Queries · User Interests · Genetic Algorithm	<p>User's interests are important in query reformation, filtering or recommender systems. Besides, the user profile is always dynamic due to the instability of his interests and the increase among documents.</p> <p>Thus, its evolution is required however identifying the changes in user's interests over time may be challenging. Recent works focused on user interests evolution, takes into account the collection by matching the document and the query or by analyzing log file of user's interactions with the system (feedbacks).</p> <p>However, they did not consider the queries exclusively in the evolution process. This paper proposes a new user profile evolution approach in which user's queries and interests are used to update the user profile.</p> <p>To do so, we use genetic algorithm technique to extract relevant interests so that to detect new interests, improve the weights of the existing ones or delete the useless among them</p>
47	ZLXC 16	A weighted adaptation method on learning user preference profile	2016	Zhiyuan Zhang, Yun Liu, Guangong Xu, Haiqiang Chen	Journal article	Elsevier	Knowledge-Based Systems	Recommender system; Numerical attributes; User preference profile; Dynamic	<p>Recommender systems typically store personal preference profiles. Many items in the profiles can be represented by numerical attributes. However, the initial profile of each user is incomplete and imprecise.</p> <p>One important problem in the development of these systems is how to learn user preferences, and how to automatically adapted update the profiles. To address this issue, this paper presents an unsupervised approach for learning user preferences over numeric attributes by analyzing the interactions between users and recommender systems.</p> <p>When a list of recommendations shown to a target user, the favorite item will be selected by him/her, then the selected item and the over-ranked items will be employed as valuable feedback to learn the user profile.</p> <p>Specifically, two contributions are offered: 1), a learning approach to measure the influence of over-ranked items through analysis of user feedbacks and 2), a weighting algorithm to</p>
48	MIM10	A Generic User Profile Adaptation Framework	2010	Lucas Marin, David Isern, Antonio Moreno	Book Chapter	IOS Press	Artificial Intelligence Research and Development	Preference; Implicit adaptation of Profile; Recommender Systems	<p>The paper presents a recommender system that permits to manage user preferences using linguistic criteria and, after collecting information about selections made by the user, it performs an unsupervised adaptation of the user profile.</p> <p>It has been implemented as a Web application and designed in a generic way so that it can be applied to any decision making problem. It includes two separate modules: a module to rate and rank all alternatives received by the system according to the current interests of the user, and a module to adapt the current user profile in an unsupervised fashion collecting implicit information about the user interaction with the system.</p> <p>The paper presents some preliminary results and discusses the performance of the adaptation algorithm.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
49	LNN+10	A Survey of Applying User Profile in the Adaptive Instructional Systems	2010	Duc-Long Le, An-Te Nguyen, Dinh-Thuc Nguyen, Van-Hao Tran, Axel	Conference article	Vietnam National University-Hochiminh city	Proceedings of the 5th International Conference on ELearning, Penang, Malaysia	adaptive system/personalized system, intelligent tutoring system, adaptive hypermedia system, eLearning, user profile	<p>Now, teaching technology with the using IT equipments has expressed its energy to improve teaching and learning in traditional learning form. In this context, computer usually becomes a powerful medium in education to support learners for searching information, doing self-study, sharing materials, communicating with friends and so on.</p> <p>It is more and more familiar and cannot miss it to all learners at all ages, and instructional systems have been also developed to meet all learning demands of everybody when working with the computer. Recently, personalized systems, called adaptive systems, have been developed in many fields such as e-Commerce, Information</p> <p>Retrieval and eLearning. In these systems, every user possesses a component of user profile. It depends on applying field that contains different information to describe him or her as personal identification, preferences, and habits. From that the system usually provides user services or information appropriate for his/her profile.</p>
50	LNNH09	Building Learner Profile in Adaptive e-Learning Systems	2009	Duc-Long Le, An-Te Nguyen, Dinh-Thuc Nguyen, Axel Hunger	Conference article	Vietnam National University-Hochiminh city	Proceedings of the 4th International Conference on e-Learning (ICEL 2009)	Adaptive e-Learning System (AeLS), learner profile, user model, profile feature, rule-based induction, α-	<p>Researches and applications of e-Learning start with building course contents and materials based on Internet, besides the development of Learning Management Systems. However, in most early e-Learning systems, learners are often provided with identical services and learning materials, in the form of "one size fits all".</p> <p>That means all learners are the same; the systems do not discriminate the learner's backgrounds, learning goals and personal interests. Now, profile is being studied to apply widely in the e-Learning systems, in which the adaptation focuses on the goal of how helping learner to get knowledge, take interest in the learning activities effectively and suitably fit to learner.</p> <p>The core idea of our paper is to suggest a generic user model that supports more teaching activities and provides appropriately learning resources and services to each learner in the blended-learning environment: traditional learning in the classroom and on-line training.</p>
51	SWM02	Adaptive user modeling for filtering electronic news	2002	M. Shepherd; C. Watters; A.T. Marath	Conference article	IEEE	Proceedings of the 35th annual Hawaii international conference on system sciences	Adaptive filters, Filtering, Artificial neural networks, System testing, Computer science,	<p>A prototype system for the fine-grained filtering of news items has been developed and a pilot test has been conducted.</p> <p>The system is based on an adaptive user model that integrates stereotypes and artificial neural networks. The stereotypes are based on newspaper sections and sub-sections, along with editor specified and user specified keywords. Eight subjects trained the system over six days of news papers (986 news items) and then tested the system on a seventh day (171 news items).</p> <p>Five users were simply asked to 'read the news' while three users developed 'corporate' profiles with explicit information needs. The evaluations suggests that such an integrated adaptive user model did, in fact, reflect the difference between the two different types of task. In both cases, the results also reflect the quality of the training of the adaptive neural network by the user in creating the user profile.</p>
52	MM14	Automatic preference learning on numeric and multi-valued categorical attributes	2014	Lucas Marin; Antonio Moreno; David Isern	Journal article	Elsevier	Knowledge-Based Systems	Recommender systems; Preference learning; Aggregation operators; Fuzzy sets; Ranking	<p>One of the most challenging tasks in the development of recommender systems is the design of techniques that can infer the preferences of users through the observation of their actions.</p> <p>Those preferences are essential to obtain a satisfactory accuracy in the recommendations. Preference learning is especially difficult when attributes of different kinds (numeric or linguistic) intervene in the problem, and even more when they take multiple possible values.</p> <p>This paper presents an approach to learn user preferences over numeric and multi-valued linguistic attributes through the analysis of the user selections. The learning algorithm has been tested with real data on restaurants, showing a very good performance.</p>
53	HYKL03	Bayesian and profile likelihood change point methods for modeling cognitive function over time	2003	Charles B.Hall; JunYing; Lynn Kuo; Richard B.Lipton	Journal article	Elsevier	Computational Statistics & Data Analysis	Change points; Longitudinal data; Mixed models; Cognitive aging; Bayesian analyses; Markov chain Monte Carlo	<p>Change point models are often used to model longitudinal data. To estimate the change point, Bayesian (Biometrika 62 (1975) 407; Appl. Statist. 41 (1992) 389; Biometrics 51 (1995) 236) or profile likelihood (Statist. Med. 19 (2000) 1555) methods may be used.</p> <p>We compare and contrast the two methods in analyzing longitudinal cognitive data from the Bronx Aging Study. The Bayesian method has advantages over the profile likelihood method in that it does not require all subjects to have the same change point. Caution must be taken regarding sensitivity to choice of prior distribution, identifiability, and goodness of fit. Analyses show that decline in memory precedes diagnosis of dementia by 7.5–8 years, and individual change points are not needed to model heterogeneity across subjects.</p>
54	Wat00	Cognitive profile analysis: A shared professional myth.	2000	Marley W. Watkins	Journal article	Guilford Publications	School Psychology Quarterly	Cognitive profiles	<p>Interpretation of cognitive test profiles is popular in school psychological training and practice but is not supported by research evidence. This special issue contains four studies that provide additional data regarding cognitive test profiles. Consistent with the largely negative research literature, detailed analysis found all four cognitive profile reports lacking reliability, validity, or diagnostic utility; even cognitive profiles composed of composites were psychometrically weak. These results were not surprising because ipsative methods are inferior to normative methods in cognitive assessment. Given this consistent failure of empirical validation, belief in the utility of cognitive test profile interpretations was likened to a shared professional myth and it was recommended that psychologists eschew the application of cognitive test profiles for differential diagnosis and remediation.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
55	GWZ H17	Collaborative Dynamic Sparse Topic Regression with User Profile Evolution for Item Recommendation	2017	Li Gao, Jia Wu, Chuan Zhou, Yue Hu	Conference article	Association for the Advancement of Artificial Intelligence	Proceedings of the AAAI Conference on Artificial Intelligence	recommender system, dynamic sparse topic modeling, user profile evolution	<p>In many time-aware item recommender systems, modeling the accurate evolution of both user profiles and the contents of items over time is essential. However, most existing methods focus on learning users' dynamic interests, where the contents of items are assumed to be stable over time.</p> <p>They thus fail to capture the dynamic changes in the item's contents. In this paper, we present a novel method CDUE for time-aware item recommendation, which captures the evolution of both user's interests and item's contents information via topic dynamics.</p> <p>Specifically, we propose a dynamic sparse topic model to track the evolution of topics for changes in items' contents over time and adapt a vector autoregressive model to profile users' dynamic interests. The item's topics and user's interests and their evolutions are learned collaboratively and simultaneously into a unified learning framework.</p> <p>Experimental results on two real-world data sets demonstrate the quality and effectiveness of</p>
56	MVC14	Combining Time Series and Clustering to Extract Gamer Profile Evolution	2014	Hector D. Menezes, Rafael Vindel, and David Camacho	Conference article	Springer	International Conference on Computational Collective Intelligence	Video-games, Gamer profile, User evolution, Time Series, Clustering	<p>Video-games industry is specially focused on user entertainment. It is really important for these companies to develop interactive and usable games in order to satisfy their client preferences. The main problem for the game developers is to get information about the user behaviour during the game-play.</p> <p>This information is important, specially nowadays, because gamers can buy new extra levels, or new games, interactively using their own consoles. Developers can use the gamer profile extracted from the game-play to create new levels, adapt the game to different user, recommend new video games and also match up users.</p> <p>This work tries to deal with this problem. Here, we present a new game, called "Dream", whose philosophy is based on the information extraction process focused on the player game-play profile and its evolution. We also present a methodology based on time series clustering to group users according to their profile evolution. This methodology has been tested with real users which have played Dream during several rounds.</p>
57	RL08	Ontology-Based User Modelling Personalization: Analyzing the Requirements of a Semantic Learning Portal	2008	Liana Razmerita & Mitiadis D. Lytras	Conference article	Springer	World Summit on Knowledge Society	Semantic Web, Databases, Personalization, Semantic Web Evolution, Semantic Web Mining, Ontology-Based User	<p>In last decade a number of design approaches have been adopted towards high performance e-learning systems. In this context Users/Learners modelling is a key milestone towards the specification of characteristics and features of learners that potentially can modify the flow of learning.</p> <p>In this article we are introducing an Ontology-Based User Modelling framework and we show how it can be used in the context of a Semantic Learning Portal.</p> <p>The main contribution of the article relates to the detailed analysis for the design variables and requirements for Semantic Learning Applications towards adoption of services, personalization and management of personal profiles and identities of learners.</p>
58	MIM13	Dynamic adaptation of numerical attributes in a user profile	2013	Lucas Marin · David Isern · Antonio Moreno	Journal article	Springer	Applied intelligence	Recommender systems · User profile · Profile adaptation · Preference learning · Numerical criteria	<p>Recommender systems try to help users in their decisions by analyzing and ranking the available alternatives according to their preferences and interests, modeled in user profiles. The discovery and dynamic update of the users' preferences are key issues in the development of these systems.</p> <p>In this work we propose to use the information provided by a user during his/her interaction with a recommender system to infer his/her preferences over the criteria used to define the decision alternatives. More specifically, this paper pays special attention on how to learn the user's preferred value in the case of numerical attributes.</p> <p>A methodology to adapt the user profile in a dynamic and automatic way is presented. The adaptations in the profile are performed after each interaction of the user with the system and/or after the system has gathered enough information from several user selections.</p> <p>We have developed a framework for the automatic evaluation of the performance of the</p>
59	AA14	Representation and Evolution of User Profile in Information Retrieval Based on Bayesian Approach	2014	Farida Achemoukh, Rachid Ahmed - Ouamer	Conference article	Springer	International Symposium on Methodologies for Intelligent Systems	Personalized Search, Short term user profile, long term user profile, Dynamic Bayesian Network.	<p>In the web personalization how to represent user profile is one of the key issues. The user profile refers to his/her interests which change over time. This paper, presents a personalized search approach for representation and evolution of the user profile, based on dynamic bayesian network.</p> <p>The theoretical framework provided by these networks allows to infer and to evolve the user profile from his /her interactions with the search system. An experimental evaluation was designed to appraise the exploitation impact of the user profile defined by his/her interests on the search results relevance.</p>
60	AS99	User Profile Modeling and Applications to Digital Libraries	1999	Giuseppe Amato & Umberto Stracciana	Conference article	Springer	International conference on theory and practice of digital libraries	Data Category Digital Library Relevance and Feedback User Information Collaborative Filter	<p>The ultimate goal of an information provider is to satisfy the user information needs. That is, to provide the user with the right information, at the right time, through the right means. A prerequisite for developing personalized services is to rely on user profiles representing users' information needs.</p> <p>In this paper we will first address the issue of presenting a general user profile model. Then, the general user profile model will be customised for digital libraries users.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
61	TCB+10	Visualizing the evolution of users' profiles from online social networks	2010	Dieudonné Tchente; Marie-Françoise Canut; Nadine Baptiste Jessel; André Pénino	Conference article	IEEE	2010 International Conference on Advances in Social Networks Analysis and Mining	user profile, social networks, temporal graph, dynamic graph, text-mining	<p>Nowadays, online social networks host more and more applications in order to provide their users with the possibility of finding everything they need on a single platform.</p> <p>The number and diversity of interactions that take place over time between users and applications within these platforms make these environments very good candidates for learning various types of information about users' interests. We are particularly interested in the determination of users' short-term and long-term interests which are essential for adaptive systems that take into account the evolution of user's needs.</p> <p>While studies in adaptive systems focus on computing interests' weight value and time periods to determine user's short-term and long-term profile, we focus instead on temporal graphs' visualization of users' interests. From a case study on Facebook, we use dynamic graphs in order to view the influence of social ties on the user's interests.</p>
62	Mo97	User Modeling in a MultiAgent Evolving System	1997	Alexandros Moukas	Conference article	Citeseer	Proceedings, workshop on Machine Learning for User Modeling, 6th International Conference on User Modeling, Chia	user modeling, multiagent system, info filtering, temporal graph's	<p>We describe Amalthea, a system that utilizes AI methods for generating and maintaining user profiles.</p> <p>The profile builder we introduce uses completely distributed learning and representation mechanisms in the form of an evolving multiagent system; it observes the current and past interaction of the user with the application as well as direct user feedback to update its state and better fulfill the interests of the user.</p> <p>Amalthea provides to its users personalized filtering and discovery of information; it's primary application domain is the World-Wide-Web and its main purpose is to assist users in finding interesting information.</p>
63	SDD08	User Profile Selection by Means of Ontology Reasoning	2008	Michael Sutterer; Olaf Droegehorn; Klaus David	Conference article	IEEE	2008 Fourth Advanced International Conference on Telecommunications	Context awareness, ontology reasoning, service delivery platform, user profile	<p>Service delivery platforms in telecommunication environments aim to host multiple services and to provide context-awareness and personalization features.</p> <p>This calls for an appropriate management of user profiles containing general user information as well as situation-dependent user preferences for the contextual personalization of these services. We present a user profile selection approach that decides on the selection of matching situation-dependent user preferences concerning the user's current situation.</p> <p>The presented approach takes advantage of ontology reasoning and is compared to other approaches. The comparison shows that our ontology reasoning based approach allows for more expressiveness concerning the specification of situation-dependent user preferences and hence leads to an added value for platform users.</p>
64	La02	The state of the art in adaptive information agents	2002	RAYMOND Y. K. LAU	Journal article	World Scientific	International Journal on Artificial Intelligence Tools	Adaptive Information Agents; Machine Learning; Belief Revision	<p>With the exponential growth of the Internet, information seekers are faced with the so-called problem of information overload. Adaptive Information Agents have been developed to alleviate this problem. The Main issues in the development of these agents are document representation, learning, and classification.</p> <p>Various paradigms have been explored for the development of adaptive information agents, and the performance of these agents differs in terms of computational efficiency, classification effectiveness, learning autonomy, exploration capability, and explanatory power.</p> <p>To develop a basic understanding of the pros and cons of these paradigms, some representative information agents are examined. Such a review also serves to identify a general for the development of the next generation adaptive information agents.</p>
65	OQP+16	Taking into Account the Evolution of Users Social Profile: Experiments on Twitter and some Learned Lessons	2016	Sirinya On-at; Arnaud Quirin; André Pénino; Nadine Baptiste Jessel; Marie-Françoise	Conference article	IEEE	2016 IEEE Tenth International Conference on Research Challenges in Information Science (RCIS)	Social profile, User profile, Egocentric network, Time-aware method	<p>Incorporating user interests evolution over time is a crucial problem in user profiling. We particularly focus on social profiling process that uses information shared on user social network to extract his/her interests. In this work, we apply our existing time-aware social profiling method on Twitter.</p> <p>The aim of this study is to measure the effectiveness of our approach on this kind of social network platform, which has different characteristics from those of other social networking sites. Although the improvement compared to the time-agnostic baseline method is still low, the experiments using a parametric study showed us the benefit of applying a time-aware social profiling process on Twitter.</p> <p>We also found that our method performs well on sparse networks and that the information dynamic influences more the quality of our proposed time-aware method than the relationships dynamic while building the social profile on Twitter. This observation will lead us to a more complex study to find out meaningful factors to incorporate user interests evolution</p>
66	YLX+08	Representation and Evolution of User Profile in Web Activity	2008	Jie Yu; Xiangfeng Luo; Zheng Xu; Fangfang Liu; Xuhui Li	Conference article	IEEE	IEEE International Workshop on Semantic Computing and Systems	Web personalization, Web service, adaptive Web site, user profile representation	<p>In the Web personalization of Web service and construction of adaptive Web site, how to represent user profile is one of the key issues. Aiming at solving the existing problems in representation of user profile includes incapability of effectively representing user's recent interest, lack in dynamically effectively updating.</p> <p>This paper presents a new method of representation of user profile. By computing the interest degree with association degree between topics, updating of interest degree is performed. By introducing memory model, it can predict which Web pages are still concerned by the user.</p> <p>By classifying the operations on the presented user profile, the construction and evolution can be done. In case study a user's browsing activities is traced and corresponding user profile is built and evolves, thus to prove the feasibility of this method.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
67	SMG06	Profile Processing and Evolution for Smart Environments	2006	Robbie Schaefer, Wolfgang Mueller & Jinghua Groppe	Conference article	Springer	International Conference on Ubiquitous Intelligence and Computing	Resource Description Framework User Preference Smart Home Preference Temperature Preference Profile	Ubiquitous systems use context information to select and adapt multimodal user interfaces and appliances for individual users in certain situations. However, in order to enable true reactive environments, context information has to be adequately collected, filtered, and processed and combined with user, device and other profiles. In this article, we present how an XML-based transcoding system can be applied for advanced profile processing and evolution. We demonstrate how to encode domain knowledge into sets of rules, which perform adaptations of user, device and context profiles for smart environments.
68	VMCG09	On the Evolution of User Interaction in Facebook	2009	Bimal Viswanath, Alan Mislove, Meeyoung Cha, Krishna P. Gummadi	Conference article	Association for Computing Machinery (ACM)	Proceedings of the 2nd ACM workshop on Online social networks	measurement online social networks	Online social networks have become extremely popular; numerous sites allow users to interact and share content using social links. Users of these networks often establish hundreds to even thousands of social links with other users. Recently, researchers have suggested examining the activity network - a network that is based on the actual interaction between users, rather than mere friendship - to distinguish between strong and weak links. While initial studies have led to insights on how an activity network is structurally different from the social network itself, a natural and important aspect of the activity network has been disregarded: the fact that over time social links can grow stronger or weaker. In this paper, we study the evolution of activity between users in the Facebook social network to capture this notion. We find that links in the activity network tend to come and go rapidly over time, and the strength of ties exhibits a general decreasing trend of activity as the social
69	CCD+19	Profiling user belief in BI exploration for measuring subjective interestingness	2019	Alexandre Chanson, Ben Crullis, Krista Drushku, Nicolas Labroche, Patrick Marcel	Conference article	HAL	DOLAP 2019	BI exploration, user belief, PageRank	This paper addresses the long-term problem of defining a subjective interestingness measure for BI exploration. Such a measure involves prior modeling of the belief of the user. The complexity of this problem lies in the impossibility to ask the user about the degree of belief in each element composing their knowledge prior to the writing of a query. To this aim, we propose to automatically infer this user belief based on the user's past interactions over a data cube, the cube schema and other users' past activities. We express the belief under the form of a probability distribution over all the query parts potentially accessible to the user. This distribution is learned using a random walk approach, and more specifically an adapted topic-specific PageRank. The resulting belief provides the foundations for the definition of subjective interestingness measures that can be used to improve the user's experience in their explorations. In the absence of ground truth for user belief, we simulate in our tests different users and their belief distributions with artificial cube explorations and evaluate our proposal
70	LN12	Monitoring User Evolution in Twitter	2012	Claudia Lauschke, Eirini Ntoutsi	Conference article	IEEE	2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining	evolution monitoring, user monitoring, Twitter	Nowadays social media are widely used for the broadcasting of different types of information, such as events, activities and opinions. Analyzing this vast amount of data for extracting models that describe individual users or groups of users has gained a lot of attention lately. In this work we analyze individual users and monitor changes in their published content over time. We propose a topic-based user profiling and monitoring approach for change detection and monitoring of profile evolution. Our method is capable of detecting persistent topics representing long term interests of the user as well as short term topics that refer to everyday events or reactions to the news. We evaluate our approach on real data from Twitter.
71	TVB09	Evolution of Context-aware User Profiles	2009	Jan Thomsen; Yves Vanroy; Yolande Berbers	Conference article	IEEE	2009 International Conference on Ultra Modern Telecommunications & Workshops	context-aware user profiles, context-awareness, adaptation, mobile computing, ubiquitous computing	Context-awareness and adaptation are key issues in mobile and ubiquitous computing. Applications on mobile devices use context information to adapt themselves to changing environments. User profiles play an important role in these systems as they serve as an individualization filter in a wide range of possible context adaptation parameters. In this paper we propose a modeling approach for the evolution of context-aware user profiles. A motivating scenario, the intelligent selection of a suitable medical expert in an emergency situation, shows the need for context-aware matching of user profiles. This is achieved by a similarity matching algorithm and reinforcement learning.
72	UB02	Learning user preferences using evolution	2002	Supiya Ujjin; Peter J. Bentley	Conference article	University College London	Proceedings of the 4th Asia-Pacific conference on simulated evolution and learning, Singapore		Recommender systems are new types of internet-based software tools, designed to help users find their way through today's complex on-line shops and entertainment websites. This paper describes a new recommender system, which employs a genetic algorithm to learn personal preferences of users and provide tailored suggestions.

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
73	LMR09	Designing user interface adaptation rules with T:XML	2009	Victor López-Jaquero, Francisco Montero, Fernando Rea	Conference article	Association for Computing Machinery (ACM)	Proceedings of the 14th international conference on Intelligent user interfaces	user interface development environment, user interface adaptation, T:XML tool.	<p>The specification of model adaptation and generation rules is a topic of great interest for the user interface development community, since there are more and more approaches supporting the model-based approach.</p> <p>The ubiquitousness in interaction and the different user profiles are not the only challenges when designing interactive systems. Furthermore, the context of use evolves over time. In this situation, there is a strong need to provide a set of adaptation rules to make the user interface evolve according to the context of use evolution.</p> <p>This paper contributes a metamodel for the definition of adaptation rules in a systematic approach, pursuing engineer adaptation. Moreover, a tool called T:XML is presented that supports the specification of adaptation rules using a visual notation that greatly simplifies the process of designing adaptation for model-based user interface environments.</p>
74	MCS08	Designing evolving user profile in e-CRM with dynamic clustering of Web documents	2008	Iraj Mahdavi; Namjae Cho; Babak Shirazi; Navid Sahebzadnia	Journal article	Elsevier	Data & Knowledge Engineering	e-CRM; Data mining; Web document clustering; Neuro-fuzzy approach; User profile	<p>Internet technology enables companies to capture new customers, track their performances and online behavior, and customize communications, products, services, and prices.</p> <p>Analyses of customers and customer interactions for electronic customer relationship management (e-CRM) can be performed by way of using data mining (DM), optimization methods, or combined approaches. One key issue in the analysis of access patterns on the Web is the clustering and classification of Web documents.</p> <p>Generally, the classification has its base on analytical models which assume a pre-fixed set of keywords (attributes) with predefined list of categories. This assumption is not realistic for large and evolving collections of documents such as World Wide Web. We propose a new approach to solve the problem of unknown number of evolving categories.</p> <p>The approach begins with the classification of test documents into a set of initial categories. A working prototype system which is based on Fuzzy Clustering CRM (FC-CRM) has been</p>
75	Bro16	Adaptive user interfaces	2016	Dermot Browne	Book	Elsevier	Google Books	Adaptive systems	
76	FS06	What Is Personalization? Perspectives on the Design and Implementation of Personalization in Information Systems	2006	Haiyan Fan & Marsha Il Scott Poole	Journal article	Taylor & Francis	Journal of Organizational Computing and Electronic Commerce	personalization, multiparadigm review, classification scheme, ideal types, design philosophy, electronic commerce	<p>In e-commerce and mobile commerce, personalization has been recognized as an important approach element in customer relationships and Web strategies.</p> <p>However, there are wide differences in how this concept is defined, characterized, and implemented in the literature. In this article we present a high-level framework for classifying approaches to personalization that delineates fundamental assumptions about personalization in the literature and relates them to strategies for developing personalization systems.</p> <p>The framework consists of 2 parts: (a) a set of perspectives on personalization that guide the design of personalization systems at a general level and (b) a scheme for classifying how personalization can be implemented.</p> <p>The personalization perspectives represent 4 distinct schools of thought on the nature of personalization distilled from the literature of several fields. These perspectives are ideal</p>
77	BBC+08	The Query-flow Graph: Model and Applications	2008	Paolo Boldi, Francesco Bonchi, Carlos Castillo, Debora	Conference article	Association for Computing Machinery (ACM)	Proceedings of the 17th ACM conference on Information and knowledge management	Query Flow Graph, Query Recommendation, Session Segmentation	<p>Query logs record the queries and the actions of the users of search engines, and as such they contain valuable information about the interests, the preferences, and the behavior of the users, as well as their implicit feedback to search engine results. Mining the wealth of information available in the query logs has many important applications including query-log analysis, user profiling and personalization, advertising, query recommendation, and more.</p> <p>In this paper we introduce the query-flow graph, a graph representation of the interesting knowledge about latent querying behavior. Intuitively, in the query-flow graph a directed edge from query q_i to query q_j means that the two queries are likely to be part of the same "search mission". Any path over the query-flow graph may be seen as a searching behavior, whose likelihood is given by the strength of the edges along the path.</p> <p>The query-flow graph is an outcome of query-log mining and, at the same time, a useful tool for it. We propose a methodology that builds such a graph by mining time and textual information as well as aggregating queries from different users. Using this approach we build</p>
78	FGRAPR23	Knowledge-Driven Profile Dynamics	2023						<p>In the last decades, user profiles have been used in several areas of information technology. In the literature, the most research works, and systems focus on the creation of profiles (using Data Mining techniques based on user's navigation or interaction history). In general, the dynamics of profiles are made by means of a systematic recreation of the profiles, without using the previous profiles. In this paper we propose to formalize the creation, representation, and dynamics of profiles in a Knowledge-Driven perspective. We introduce and axiomatically characterize four operators for changing profiles using a belief change inspired approach.</p>

N	ID	Title	Year	Author	Type	Publisher	Published in	Keywords	Abstract
79	LL09	Deriving Concept-Based User Profiles from Search Engine Logs	2009	Kenneth Wai-Ting Leung; Dik Lun Lee	Journal article	IEEE	IEEE Transactions on Knowledge and Data Engineering	Negative preferences, personalization, personalized query clustering, search engine, user profiling.	User profiling is a fundamental component of any personalization applications. Most existing user profiling strategies are based on objects that users are interested in (i.e., positive preferences), but not the objects that users dislike (i.e., negative preferences). In this paper, we focus on search engine personalization and develop several concept-based user profiling methods that are based on both positive and negative preferences. We evaluate the proposed methods against our previously proposed personalized query clustering method. Experimental results show that profiles which capture and utilize both of the user's positive and negative preferences perform the best. An important result from the experiments is that profiles with negative preferences can increase the separation between similar and dissimilar queries. The separation provides a clear threshold for an agglomerative clustering algorithm to terminate and improve the overall quality of the resulting query clusters.
80	GKVL+07	Creating an Ontology for the User Profile: Method and Applications	2007	Golemata Maria, Katifori, Vassilakis Costas, Lepouras Georgios, Halatsis	Conference article	Helios	Proceedings AI* AI Workshop RCIS	user profile, ontology, user modeling, context	User profiling is commonly employed nowadays to enhance usability as well as to support personalization, adaptivity and other user-centric features. Insofar, application designers model user profiles mainly in an ad-hoc manner, hindering thus application interoperability at the user profile level, increasing the amount of work to be done and the possibility of errors or omissions in the profile model. This work aims at creating a user profile ontology that incorporates concepts and properties used to model the user profile. Existing literature, applications and ontologies related to the domain of user context and profiling have been taken into account in order to create a general, comprehensive and extensible user model. This ontology can be used as a reference model, in order to alleviate the aforementioned issues. The model, available for download, is exemplified through its application in two different areas, personal information management and adaptive visualization.
81	ENSN19	A Survey of User Profiling: State-of-the-Art, Challenges, and Solutions	2019	Christopher Ifeanyi Eke; Azah Anir Norman; Liyana Shuib; Henry Friday Nweke Computer	Journal article	IEEE	IEEE Access	User profiling, user interest, profiling modeling, personalized service, service recommendation.	Advancements in information and communication technology, and online web users have given attention to the virtual representation of each user, which is crucial for effective service personalization. Meeting users need and preferences is an ongoing challenge in service personalization. This issue can be addressed through the building of a comprehensive user profile. A user profile is the summary of the user's interests, characteristics, behaviours, and preferences, while user profiling is the system of collecting, organizing and inferring the user profile information. Many reviews on user profiling have been conducted but none focused on the effective profile modeling process. Hence, this article aims to provide a review of the recent state-of-the-art approach to user profiling. These include methods, description, characteristics, and taxonomy of the user profile. The study of the existing user profiling modeling in the aspect of data acquisition, feature extraction, profiling techniques, and profiling approaches (with the identification of their strengths and weaknesses) and the performance measures are also provided. In addition, the research challenges were also discussed with a focus on privacy, datasets, cold start issues, trust issues, and computational complexity. Moreover, the article identified an open research direction that serves as
82	TYZZ10	A combination approach to web user profiling	2010	Jie Tang, Limin Yao, Duo Zhang, Jing Zhang	Journal article	Association for Computing Machinery (ACM)	ACM Transactions on Knowledge Discovery from Data (TKDD)	User profiling, information extraction, name disambiguation, topic modeling, social network, text mining	In this article, we study the problem of Web user profiling, which is aimed at finding, extracting, and fusing the "semantic"-based user profile from the Web. Previously, Web user profiling was often undertaken by creating a list of keywords for the user, which is (sometimes even highly) insufficient for main applications. This article formalizes the profiling problem as several subtasks: profile extraction, profile integration, and user interest discovery. We propose a combination approach to deal with the profiling tasks. Specifically, we employ a classification model to identify relevant documents for a user from the Web and propose a Tree-Structured Conditional Random Fields (TCRF) to extract the profile information from the identified documents; we propose a unified probabilistic model to deal with the name ambiguity problem (several users with the same name) when integrating the profile information extracted from different sources; finally, we use a probabilistic topic model to model the extracted user profiles, and construct the user interest model. Experimental results on an online system show that the combination approach to different profiling tasks clearly outperforms several baseline methods. The extracted profiles have been applied to expert finding, an important application on the Web. Experiments show that the accuracy of expert
83	AHX13	A survey of user modelling in social media websites	2013	Abdel-Hafez, Ahmad & Xu, Yue	Journal article	Canadian Center of Science and Education	Computer and Information Science	user modelling, social media, semantic enrichment, and dynamic user modelling	With the widespread of social media websites in the internet, and the huge number of users participating and generating infinite number of contents in these websites, the need for personalisation increases dramatically to become a necessity. One of the major issues in personalisation is building users' profiles, which depend on many elements; such as the used data, the application domain they aim to serve, the representation method and the construction methodology. Recently, this area of research has been a focus for many researchers, and hence, the proposed methods are increasing very quickly. This survey aims to discuss the available user modelling techniques for social media websites, and to highlight the weakness and strength of these methods and to provide a vision for future work in user modelling in social media websites.
84	Kru97	LIFESTYLE FINDER: Intelligent User Profiling Using Large-Scale Demographic Data	1997	Bruce Krulwich	Journal article	PKP Publishing Services Network	AI magazine		A number of approaches have been advanced for taking data about a user's likes and dislikes and generating a general profile of the user. These profiles can be used to retrieve documents matching user interests; recommend music, movies, or other similar products; or carry out other tasks in a specialized fashion. This article presents a fundamentally new method for generating user profiles that takes advantage of a large-scale database of demographic data. These data are used to generalize user-specified data along the patterns common across the population, including areas not represented in the user's original data. I describe the method in detail and present its implementation in the LIFESTYLE FINDER agent, an internet-based experiment testing our approach on more than 20,006 users worldwide.

D.2 Litmaps articles

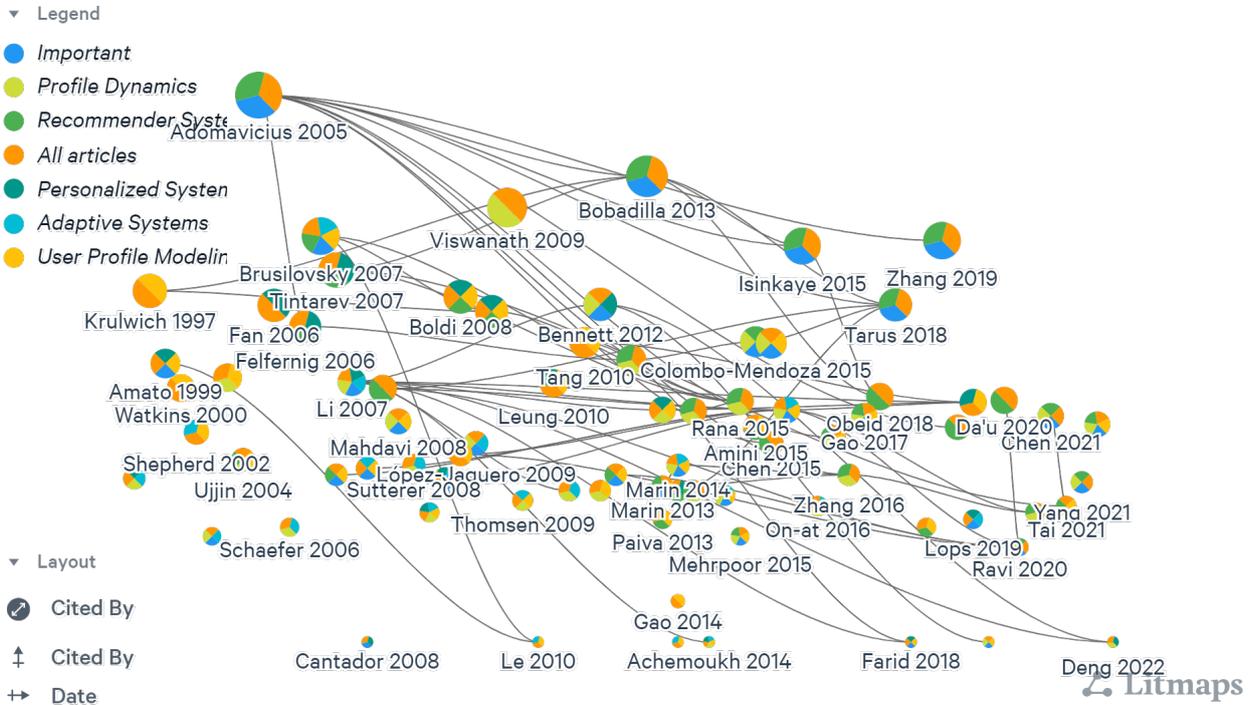


Figure D.1: All articles

APPENDIX D. ARTICLE DATABASE

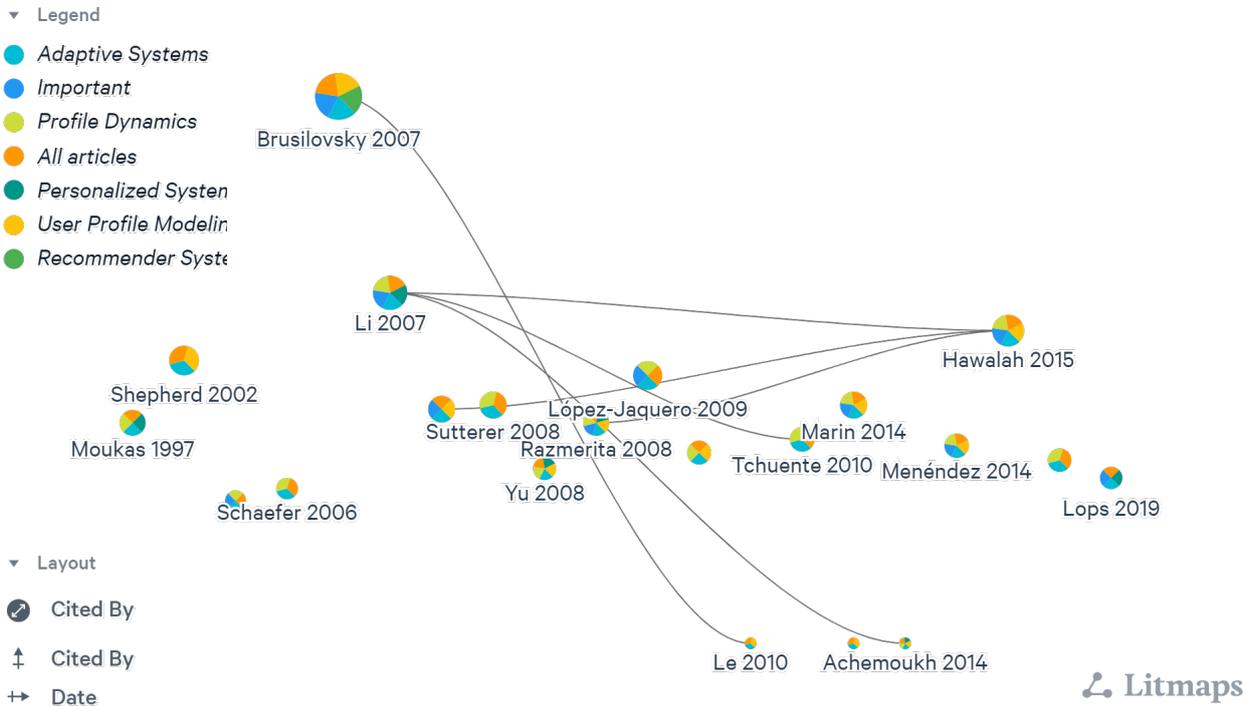


Figure D.2: Adaptive systems

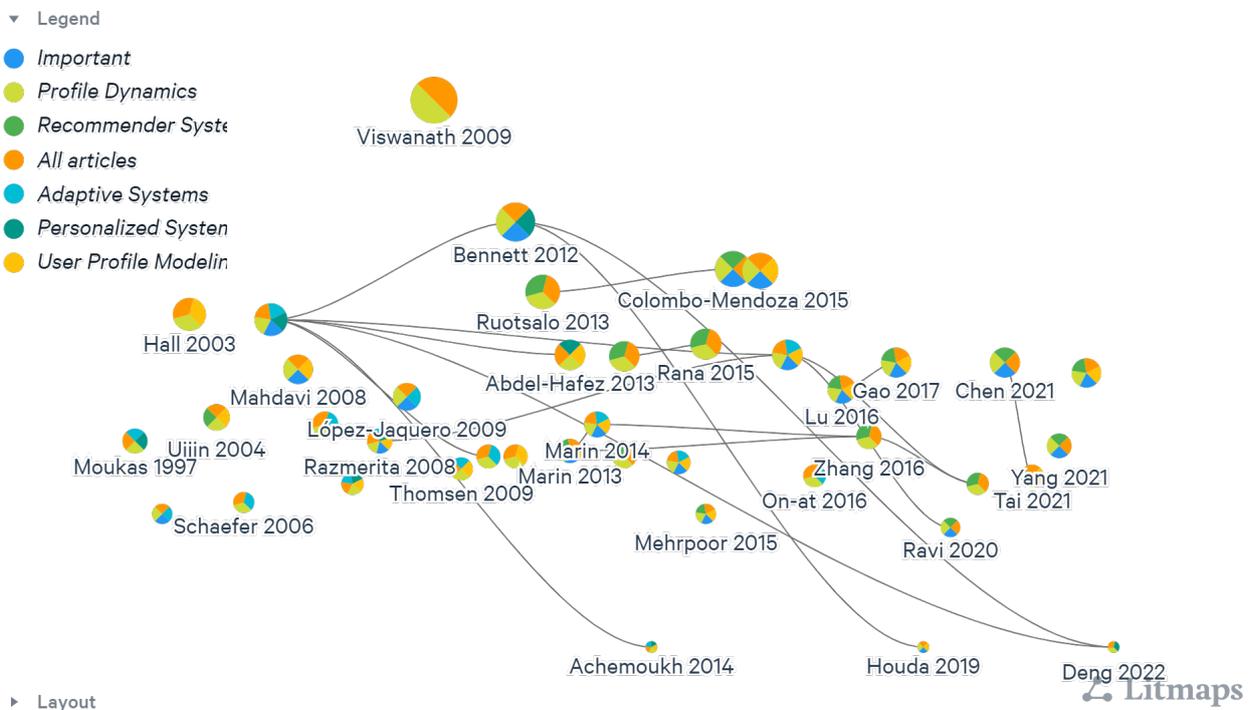


Figure D.3: Profile Dynamics

APPENDIX D. ARTICLE DATABASE

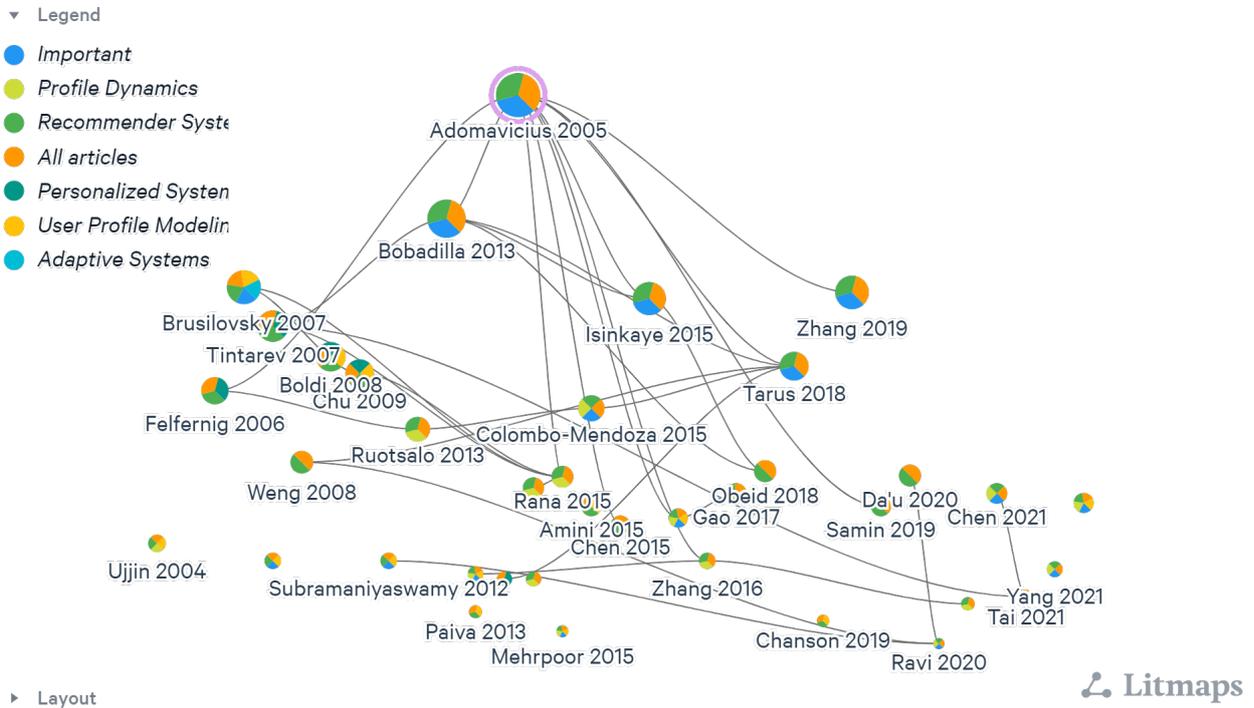


Figure D.4: Recommender Systems

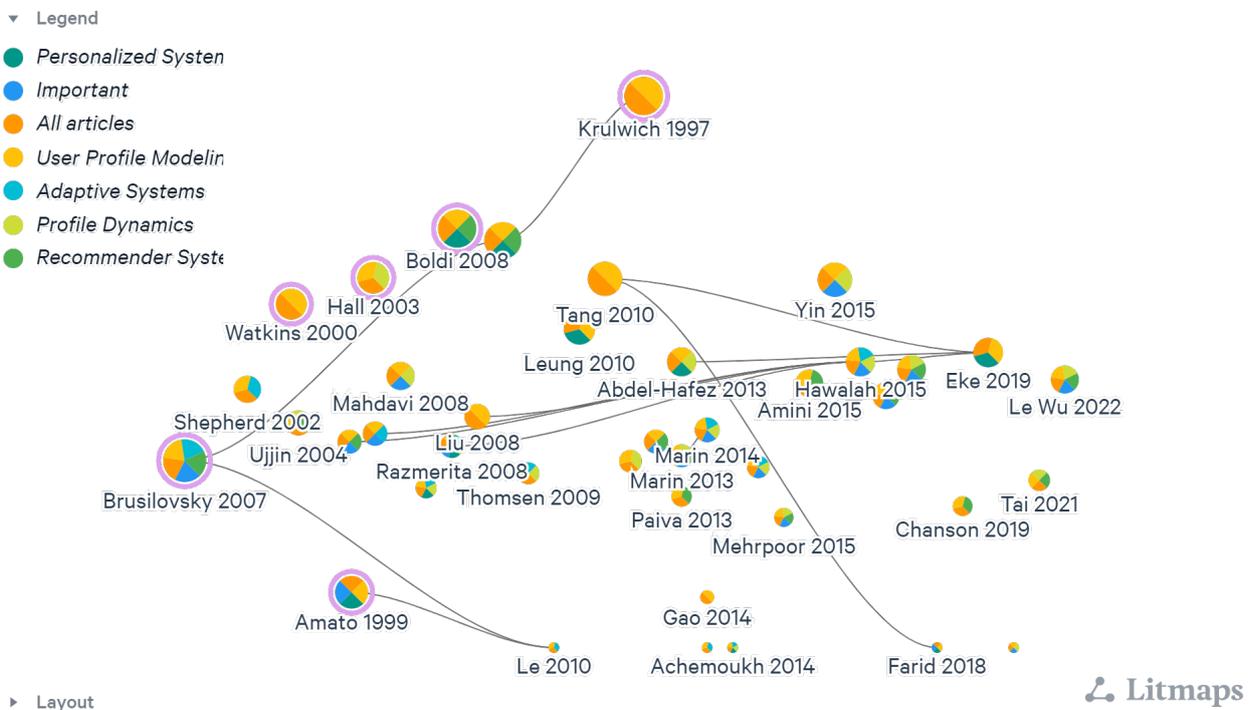


Figure D.5: User Profile Modeling

APPENDIX D. ARTICLE DATABASE

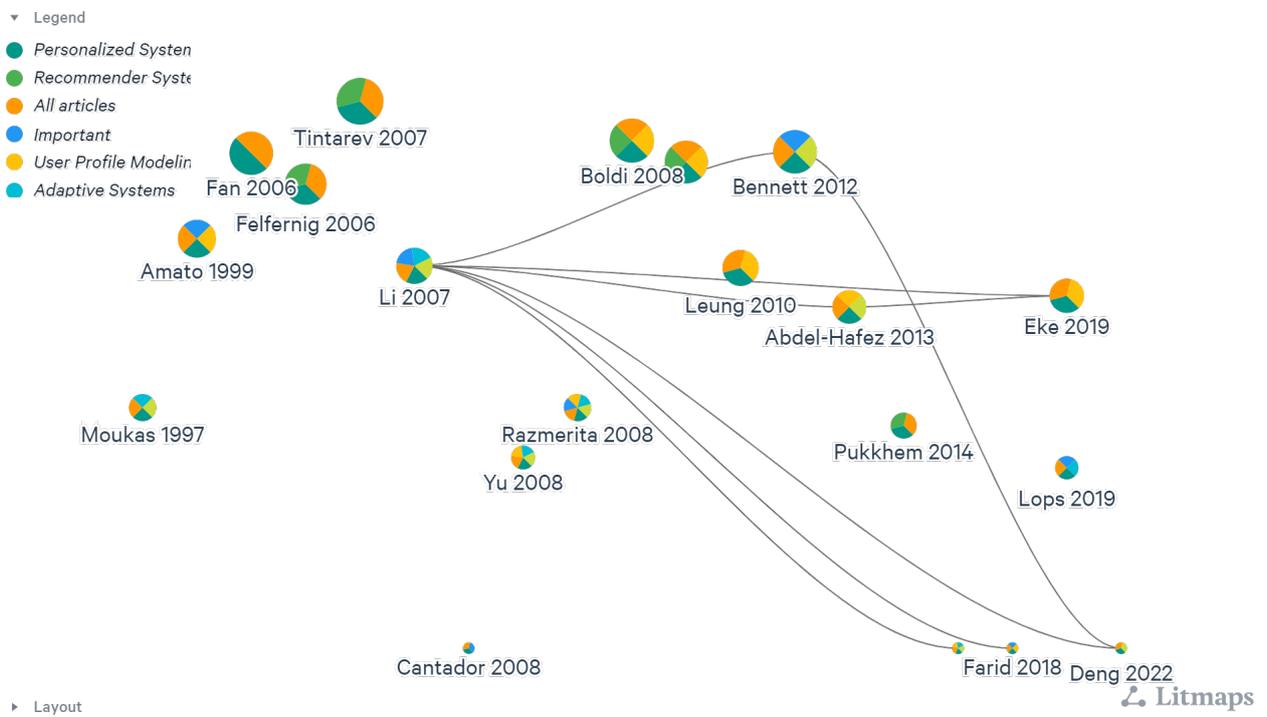


Figure D.6: Personalized Systems