SELECTIVE PERSONALIZATION USING TOPICAL USER PROFILE TO IMPROVE SEARCH RESULTS

ARAMA SONUÇLARINI İYİLEŞTİRMEK İÇİN KONU KULLANICI PROFİLİNİ KULLANARAK SEÇİCİ KİŞİSELLEŞTİRME

SAMIRA KARIMI MANSOUB

PROF. DR. İLYAS ÇİÇEKLİ
Supervisor

Submitted to
Graduate School of Science and Engineering of Hacettepe University
as a Partial Fulfillment to the Requirements
for the Award of the Degree of Doctor of Philosophy
in Computer Engineering

2020

This thesis is dedicated to the memory of my beloved mother who passed away before I started my doctoral studies. She taught me to be a powerful person in life...

And to Jean-Paul Sartre who said "My thought is me: that is why I cannot stop thinking."

ABSTRACT

SELECTIVE PERSONALIZATION USING TOPICAL USER PROFILE TO IMPROVE SEARCH RESULTS

Samira KARIMI MANSOUB

Doctor of Philosophy, Computer Engineering Department

Supervisor: Prof. Dr. İlyas ÇİÇEKLİ

Co- Supervisor: Asst. Prof. Dr. Gönenç ERCAN

October 2020, 150 pages

Personalization is a technique used in Web search engines to improve the effectiveness of information retrieval systems. In the field of personalized web search has recently been doing a lot of research and applications. In this research, we evaluate the effect of personalization for queries with different characteristics. With this analysis, the question of whether personalization should be applied for all queries in the same way or not is investigated. While personalizing some queries yields significant improvements on user experience by providing a ranking inline with the user preferences, it fails to improve or even degrades the effectiveness for less ambiguous queries. A potential for personalization metric can improve search engines by selectively applying a personalization.

Current methods for estimating the potential for personalization such as click entropy and topic entropy are based on the clicked document for query or query history. They have limitations like unavailability of the prior clicked data for new and unseen queries or queries without history. In this thesis, the topic entropy measure is improved by integrating the user distribution to the metric, robust to the sparsity problem. This metric estimates the potential for personalization using a topical user profile created on user documents. In this way, we can overcome the cold start problem to estimate the potential for new queries and increase the accuracy of estimates for queries with history.

Although in this thesis the main focus is on topic-based user profiles, since there is not more research on keyphrase-based user profiles in the process of personalization, we do a comparison research between keyphrase-based and topic-based profiles. We examine how personalization can be integrated into the state of the art keyphrase extraction models by considering different models of supervised and unsupervised methods. We evaluate topic-based and keyphrase-based user profiles using a re-ranking algorithm to complete the process of personalization using different datasets. In personalization using keyphrase-based profiles, personalized models based on supervised keyphrase extraction approaches obtained more accuracy by 7% than unsupervised approaches however it does not improve compared to topic-based models.

In topic-based models, we use a combination of personalization in the level of user-specified and group profiling as part of the ranking process. In the previous ranking methods, more improvement in ranking is for the queries which match the user's history. To take advantage of ranking for all queries, we present a group personalized topical model(GPTM) that uses groups obtained from clustered similar users on topical profiles. Experiments reveal that the proposed potential prediction method correlates with human query ambiguity judgments and group profiles based ranking method improve the Mean Reciprocal Rank by 8%.

Keywords: User Search Behavior, Topic-based User Profile, Keyphrase-based User Profile, Personalized Web Search, Latent Dirichlet Allocation

ÖZET

ARAMA SONUÇLARINI İYİLEŞTİRMEK İÇİN KONU KULLANICI PROFİLİNİ KULLANARAK SEÇİCİ KİŞİSELLEŞTİRME

Samira KARIMI MANSOUB

Doktora, Bilgisayar Mühendisliği

Danışman: Prof. Dr. İlyas ÇİÇEKLİ

Eş Danışman: Dr. Öğr. Üyesi Gönenç ERCAN

Ekim 2020, 150 sayfa

Kişiselleştirme tekniği, bilgi erişim sistemininin etkinliğini iyileştirmek için web arama motorlarında kullanılan bir tekniktir. Kişiselleştirilmiş web araması alanında son günlerde sıkça araştırma ve uygulama yapılmaktadır. Bu araştırmada öncelikle farklı karakteristiklere sahip sorgularda kişiselleştirmenin etkisini ölçüyoruz. Bu analiz ile kişiselleştirmenin tüm sorgulara aynı biçimde uygulanıp uygulanmayacağı araştırılmaktadır. Bazı sorguların kişiselleştirilmesi, kullanıcı tercihleri ile aynı öncelikte bir derecelendirme yapıldığı takdirde kullanıcı deneyiminde önemli iyileştirmeler sağlarken, belirsiz sorguların etkinliğini iyileştirmez ve hatta bazen azalmasına neden olur. Kişiselleştirme metriklerinin potansiyeli, kişiselleştirmeyi seçilimli bir şekilde uygulayarak arama motorlarını iyileştirebilir. Kişiselleştirme potansiyelini tahmin etmek için kullanılan ”tıklama entropisi” ve ”konu entropisi” gibi mevcut yöntemler, sorgu ve sorgu geçmişi için tıklanan dokümanlara dayalıdırlar. Bu yöntemlerin, önceki tıklanmış verilerin kullanışsızlığı gibi yeni ve görünmeyen sorgular veya geçmişi olmayan sorgular için bir takım sınırları mevcuttur.

Bu tezde, konu entropisi ölçümü, kullanıcı dağılımını metriğe entegre ederek seyreklik problemine dayanıklı olacak şekilde iyileştirildi. Bu metrik kullanıcı dökümanları üzerinde oluşturulan konusal kullanıcı profillerini kullanarak kişiselleştirme potansiyelini tahmin eder. Bu yolla yeni sorguların potansiyelini tahmin etmek ve geçmişe sahip sorguların doğruluğunu artırmak için soğuk başlangıç probleminin üstesinden gelebiliriz.

Bu tezde her ne kadar ana odak noktası konu tabanlı kullanıcı profilleri olsa da anahtar kelime tabanlı kullanıcı profilleri hakkında çok araştırma olmadığı için, anahtar kelimeye dayalı profiller ve konu tabanlı profiller arasında bir karşılaştırma yapıyoruz. Denetimli ve denetimsiz yöntemlerin farklı modellerini göz önünde bulundurarak, kişiselleştirmenin ustalık derecesinde anahtar kelime çıkartma modellerine nasıl entegre edilebileceğini inceliyoruz. Farklı veri kümelerini kullanarak kişiselleştirme sürecini tamamlamak için yeniden derecelendirme algoritmaları kullanılarak anahtar kelime ve konu tabanlı kullanıcı profillerini değerlendiriyoruz. Anahtar kelime tabanlı kişiselleştirmede, denetimli anahtar kelime çıkartma yaklaşımı %7 oranında iyileştirme sağlarken, konu tabanlı kişiselleştirmede bir iyileştirme sağlamamaktadır.

Konu tabanlı modellerde, derecelendirme sürecinin bir bölümü olarak, kullanıcı tarafından belirlenen ve grup profili seviyesinde kişiselleştirme kombinasyonu kullanırız. Önceki derecelendirme yöntemlerinde iyileştirmeler, kullanıcının geçmişi ile eşleşen sorgular içindir. Derecelendirmenin avantajlarını tüm sorgularda kullanmak için, konusal profiller üzerinde kümelenen benzer kullanıcılardan elde edilmiş grupları kullanan grup kişiselleştirmiş konusal modelini (GPTM) tanıtıyoruz. Deneyler, potansiyel tahmin yönteminin insan sorgularında belirsiz kararlar ile ilişkili olduğunu ve grup tabanlı derecelendirmenin ortalama karşılıklı derecelendirmeyi %8 artırdığını göstermektedir.

Anahtar Kelimeler: Kişiselleştirilmiş Arama, Kullanıcı Profili, Kullanıcı Arama Davranışı, Topikal Kullanıcı Modeli

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my supervisors Asst. Prof. Dr. Gönenç Ercan and Prof. Dr. İlyas Çiçekli, for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. They have always been there for me to provide encouragement and helpful feedback on my research.

Besides my supervisors, I would like to thank the rest of my thesis committee members, Prof. Dr. Pınar Karagöz, Asst. Prof. Dr. Burcu can, for their insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives.

Last but not least, I would like to thank my family, especially my dearest husband Dr. Rayan Abri, who has given me the greatest motivation in finishing my work. To be honest, I would never have completed a Ph.D. without him. I sincerely appreciate his belief in me.

CONTENTS

Page

TABLES

FIGURES

ABBREVIATIONS

LDA	**L**atent **D**irichlet **A**llocation
TOCHI	**T**ransactions **O**n **C**omputer-**H**uman **I**nteraction
pLSA	**p**robabilistic **L**atent **S**emantic **A**nalysis
KL Divergence	**K**ullback-**L**eibler **D**ivergence
PLSI	**p**robabilistic **L**atent **S**emantic **I** ndexing
CASPER	**CAS**e-based **P**rofiling for **E**lectronic **R**ecruitment
RR	**R**eciprocal **R**ank
MRR	**M**ean **R**eciprocal **R**ank
TUE	**T**opic **U**ser **E**ntropy
UTUE	**U**nified **T**opic **U**ser **E**ntropy
KEA	**K**eyphrase **E**xtraction **A**lgorithm
PTM	**P**ersonalized **T**opic **M**odel
PKM	**P**ersonalized **K**eyword/keyphrase **M**odel
NonPTM	**N**on-**P**ersonalized **T**opic **M**odel
GPTM	**G**rouped **P**ersonalized **T**opic **M**odel
NDCG	**N**ormalized **D**iscounted **C**umulative **G**ain
CE	**C**lick **E**ntropy
TE	**T**opic **E**ntropy
NCE	**N**ormilized **C**lick **E**ntropy
UE	**U**ser **E**ntropy
HTML	**H**yper **T**ext **M**arkup **L**anguage
ODP	**O**pen **D**irectory **P**roject
VSM	**V**ector **S**pace **M**odel
SVM	**S**upport **V**ector **M**achine
KNN	**K**-**N**earest **N**eighbour
NN	**N**eural **N**etworks

RDF	**R**esource **D**escription **F**ramework
TF-IDF	**T**erm **F**requency **I**nverse **D**ocument **F**requency
ID3	**I**terative **D**ichotomiser 3
GUMO	**G**eneral **U**ser **M**odel **O**ntology
AOL	**A**merican web portal and **O**n**L**ine service
RAKE	**R**apid **A**utomatic **K**eyword **E**xtraction
PKE	**P**ython-based **K**eyphrase **E**xtraction
BM25	**B**est **M**atching 25
IR	**I**nformation **R**etrieval
MRR	**M**ean **R**eciprocal **R**ank
MAP	**M**ean **A**verage **P**recision

1. INTRODUCTION

1.1. Motivation

The increased volume of information and the high demand for information retrieval from web-based systems motivate research related to personalization methods. Personalized services are common in web search, recommendation, filtering systems, and other adaptive websites that can tailor themselves concerning users' needs. Personalizing web search by re-ranking the retrieved documents concerning a user's interests is adopted by many search engines. The personalization of broad and ambiguous queries yield a better user experience. For instance, for the query "*test*", if the user issuing the query is a medical professional, results relevant to medical tests should be preferred over tests for evaluating students in educational institutes. On the other hand, for other queries with a more clear and specific meaning, the ranking methods without any personalization are more effective [1]. A measure able to estimate the potential for personalization can enable the selective application of personalization and improve the overall effectiveness of the search system.

According to some research, personalization should not be used for all queries in the same way because it varies in effectiveness for different queries. There is a need to estimate the potential for personalization for queries. Exploring the characteristics of a query is necessary to achieve a correct personalization process. For this purpose, query features will investigate in several aspects including query structure, query history, the clicked results for each query. Then we will investigate the effective features for estimating the potential of queries.

Different measures are used to determine the potential for personalization of queries [2, 3]. Click entropy measured using the query history and documents clicked by the users is one such measure [2]. This method is recently improved by a topic model-based extension [3] and referred to as topic entropy. Although these metrics can estimate the potential for personalization partly, they have limitations especially for queries without history. For these queries, as no document is previously clicked estimating their click and topic entropy is impossible. To overcome this problem, a new metric called unified topic entropy is proposed, which estimates the potential for personalization using the topical user profile created on the clicked document by the user for other queries. In this thesis, we improve topic entropy by measuring how each user's topical user profile differentiates from the query words' topics. Using the topic distributions of clicked documents for each user as a feature, the potential for personalization is modeled on a fine-grained level.

Through experiments, we show that the proposed method can process queries without any history and is more effective for queries with low frequency. This allows the system to overcome the cold-start problem.

Along with estimating the potential for personalization for queries, it is needed to create a user profile. The user profiles are created using user-specific content, user behavior, and user context. To obtain contextual and background information about the user, we should follow the steps such as collecting, processing, and analyzing information for creating user profiles. The structure of the user profile always is an important issue because of its impact on ranking performance. It is clear that if the algorithms used in the user model be more accurate and powerful, the user model and personalized services will result in better efficiency and quality. Therefore, creating an efficient user profile is a challenge.

In this thesis, we consider both user contents such as the features extracted from queries and user behaviors such as clicked documents. Our purpose is to show the importance of hidden topics in the clicked documents by users. To do this, we create the topical user profile for each user using the clicked documents. Besides, we conduct comparative research on keyword and keyphrase-based user-profiles intending to consider keyphrase-based user profiles in the personalization process. Our motivation is to develop a Keyphrase-based profile that operates on documents to improve in the process of personalization.

Finally, a ranking method completes the process of personalized search. Topic models have also used in the process of ranking the search results [4, 5]. There is the sparsity problem for user profiles when a user does not have sufficient history. To resolve this problem, the users are first grouped as the latent topics modeled using Latent Dirichlet Allocation (LDA). Our proposed ranking model re-ranks the search results concerning user group profiles instead of individual user profiles. The proposed group profile improves retrieval effectiveness when using both long-term and short-term query histories.

In summary, this thesis proposes two novel algorithms concerning two sub-tasks of web search personalization. The first algorithm focuses on estimating the potential for personalization using our proposed metric which is referred to as unified topic user entropy. Second is a re-ranking method for personalization using grouped topic profiles. When these two contributions are used in combination, a clear improvement over the baseline methods is achieved.

1.2. Contribution and Outline

1.2.1. Main Contributions

The contributions of the thesis are as bellows:

- To estimate the potential for personalization for queries, exploring the characteristics of the query to achieve a correct personalization process. For this purpose, query features are investigated in several aspects.

- In this framework, a new metric is proposed using the topic distribution of user documents in a topical user profile, to estimate the potential for personalization for all queries. In this way, we can overcome the cold start problem to estimate the potential for new and unseen queries and increasing the accuracy of estimates for queries with history. The purpose is to prevent useless personalization by identifying the appropriate queries for personalization and filter out unappropriated queries such as navigational queries. The new metric will be applied in the process of ranking as a weighting approach to estimate the potential for personalization.

- After estimating the potential for personalization for queries, it is needed to create a user profile using user behavior. To create a user profile, we follow the steps such as collecting, processing, and analyzing information. In this thesis, we consider both user contents such as the features extracted from queries and user behaviors such as clicked documents.

- In the process of creating the user profiles, keyword/keyphrase-based and topic-based user profiles will be applied to represent user interest. Topic-based profiles are created using the LDA model on query log which is divided into short-term and long-term to consider user interest in different time intervals to compare efficiency.
- Finally, ranking methods will be applied to built user profiles to improve personalized search results. To achieve more improvements selective topic models of personalized and non-personalized models are proposed in this thesis. To take advantage of ranking for all queries, combination methods will be used which are using topical user profiles and group profiles. Topical group profiles are created by clustering on the topical user profile.

1.2.2. Thesis Outline

An outline of the chapters in this thesis is including an overview of the related works of need for personalization, search personalization, and user modeling, that are discussed in detail. Following that, three chapters are detailing our contributions and explain the main works related to providing a new model to estimate personalization, user modeling, and re-ranking to improve personalized search. The organization of the thesis is as follows:

Chapter 3. discusses the related work on user profiling as topic-based models and keyword/keyphrase based models and personalized search approaches. It is focused on describing the personalized web search using user profiles as topic-based and keyword/keyphrase based profiles.

The method including the main phases as an overview is discussed in chapters 4., 5. and 6. These chapters include main phases estimating the potential for personalization, creating a user profile and ranking the results. In chapter 4., estimating the potential for personalization for a global query will be considered. In this chapter, a new metric called unified topic user entropy is presented as a solution to estimate the potential for new/unseen queries or queries without history.

In chapter 5., user profiles are built. In this chapter, keyword, and key phrase-based and topic-based user profiles will be created to represent user interest. In the main part of the research, in this chapter topic-based profiles are created using the LDA model on query log which is divided into short-term and long-term to consider user interest in different time intervals to compare efficiency.

Finally, in the last part of the method in chapter 6., ranking methods will be applied to improve personalized search results. This chapter presents our new proposed personalization search method based on grouped user profiles. The evaluation methodology is given in Chapter 7. We perform the experiments to study the effectiveness of the ranking methods with presented models and evaluation results are presented at the end of this chapter. Chapter 8. includes the concluding and future works remarks.

2. BACKGROUND INFORMATION

Personalization depends on prior knowledge about information retrieval and web search. Personalization aims to build accurate and detailed user models, thus a definition of what is meant by the user model is introduced first. Both the contexts of what type of data is used in user-profiles and the models to represent them are introduced. User models in the form of keyword-based, topic-based, semantic-based, rule-based, and ontology-based representations are defined for completeness, however, we will concentrate on keyword/keyphrase-based and topic-based that are under-explored in the literature. Given a user model, personalization methods are used to change the behavior of ranking or recommendation algorithms. Integrating the user model to generic retrieval systems require machine learning algorithms. The techniques used in machine learning and predictive statistical methods are presented in this Chapter as required background information.

Since the problem is using personalized search in the process of re-ranking, it is necessary to define the notion of retrieval, query, and re-ranking. A retrieval system first gets a query from a user and then re-ranks the results concerning her/his user profile. In this process, a query can consist of a string of different lengths. A re-ranking method re-orders and generates a list of documents returned by the search engine for the given query using the user profile. In our research, a generic ranking method refers to a re-ranking method that uses a generic document scoring function based on topic models without any personalization, while a personalized re-ranking method uses the personalization factor for the created user profile using the documents clicked.

2.1. Estimating Potential for Personalization

In this section, the potential for personalization is defined and common techniques used for search personalization are discussed.

2.1.1. Potential for Personalization

In the process of personalized search, personalization is not appropriate for all user queries and may even yield worse results than generic ranking methods. The ranking for a navigational, specific, and unambiguous query is usually stable and its ranking does not depend on the user preferences. Better rankings can be obtained for those queries without personalization. For example, the query "my space" is usually a navigational query for the social networking website regardless of the user issuing this query. For such a query, trying to personalize the results can produce an inferior ranking. For such queries, it seems that personalization does not result in improvement.

Thus it is important to quantify how much a query can benefit from personalization. Teevan et al. [6] defines the potential for personalization as the gap between the results returned by the search engine to everyone (generic ranking) and the tailored and expected result to satisfy an individual. In this chapter, we discuss the metrics used to estimate the potential, and in chapter 4. we will estimate the potential for personalization for all queries in the data set.

2.1.2. Estimating Potential Techniques and Methods

A query is an element to represent the user's goals(from search) by the search engine. Given this query, predicting its potential for personalization is an important task with a direct impact on the final effectiveness of the system. Teevan et al. [6, 7] evaluated different metrics to predict the ambiguity of a query and its potential for personalization. They evaluated intrinsic features such as query length, click entropy introduced by Duo et al. [2], clarity measure which compares the language model of the retrieved result set to a background language model [8] and result entropy for predicting the potential for personalization.

Click entropy

In the research conducted by Teevan [6, 7], the potential for queries was investigated, and click entropy is introduced as a dominant indicator. Click entropy measures the variability in clicked results across individuals as Equation 1.

$$ClickEntropy(q, D_q) = \sum_{d \in D_q} -P(d|q)log(P(d|q)) \tag{1}$$

Where D_q is the set of documents clicked for the query q and $P(d|q)$ is the number of clicks for a document d divided by the total number of clicks for the query q. In more research, the click entropy is normalized to [0, 1] as follows:

$$NormalizedClickEntropy(q, D_q) = \frac{ClickEntropy(q, D_q)}{log_2|D_q|} \tag{2}$$

Click-entropy models the ambiguity using only the user interactions, ignoring the contents of the documents.

User entropy

The user entropy is also another indicator of entropy that averages click entropy by each user. Wang et al. [9] proposed user entropy and discussed that the user entropy is useful for low-frequency queries. They reported click entropy as a reliable method for predicting the potential when the history for the query is available. User entropy can be calculated as Equation 3.

$$UserEntropy(q, U_q, D_q) = \frac{\sum_{u \in U_q} ClickEntropy(q, D_{u,q})}{|U_q|} \tag{3}$$

Where $D_{u,q}$ is the set of documents clicked for the query q by user u and $|U_q|$ is the number of users that submitted query q.

Topic entropy

Song et al. [10] discussed the relationship between query ambiguity and topic distributions. They used the latent topic model variable to model the clicked documents' content and improved the click-entropy model for predicting the ambiguity of queries. Topic entropy discussed by Yano [3] models $P(d|q)$ using the topic model distribution of the documents, able to account for documents with similar contents. It can be calculated using Equation 5.

$$TopicEntropy(q, D_q) = \sum_{d \in D_q} P(d|q) KL(P(z|d)||P(z|q)) \tag{4}$$

$$= \sum_{d \in D_q} P(d|q) \sum_{z \in Z} P(z|d) log(\frac{P(z|d)}{P(z|q)}) \tag{5}$$

Where $P(z|d)$ is the probability of the topic z for the given document d and $KL(P(z|d)||P(z|q))$ is Kullback-Leibler Divergence between two probability distributions $P(z|d)$ and $P(z|q)$. In this thesis, the topic set Z is obtained using Latent Dirichlet Allocation (LDA). $P(z|q)$ is the probability of the topic z or the given query q and it is estimated using the documents clicked for a query q as in Equation 6.

$$P(z|q) = \sum_{d \in D_q} P(z|d) P(d|q) \tag{6}$$

2.2. Generic Forms of User Models

Since there is not a classification on user models for research and the need for such research is strongly felt, so we first discuss the generic forms of user models or the types of context incorporated in the user models and then we are going to classify the major dimensions that play a critical role in user modeling. We discuss in detail based on conducted research in the next chapter. These dimensions are individual or group profiling, explicit or implicit feedback, supervised and unsupervised learning, short-term or long-term interests, distributive or non-distributive user models and dynamic or static profiling.

2.2.1. Context Incorporated in a User Model

This chapter explains the types of context incorporated in a user model. Although in initial user models, personal and background information such as name, age, address, email, phone number was the main basic in creating user models but there is also some complex context information as critical elements in creating user model. The incorporated context in the structure of a user model can be a different range such as user interests, preferences, knowledge, abilities, background, goals, skills, individual traits depending on the behind intent in creating user model.

In addition to some of the contexts can share some cases and ideas depending on the purpose of the application. In many of personalization, the data about user behavior and interactions between users and systems is important to collect. Besides, in more research, user models can be a combination of different contexts listed. For instance authors in [11] created a user model with several parameters including personal information such as user identity, name, address, etc., general characteristics such as weight, height and physical abilities, state of the user such as education, occupation, expertise, and user capabilities and preferences.

Because the interests and priorities of the user are changing, a comprehensive user model must adapt itself with different variations. So the correct choice of different combinations of contexts to create an efficient user model is very important. There is a general division based on feature-based and stereotype models. Based on the type of feature incorporated, it can be distinguished the user model into different categories. These features can be static or variable. We consider both models, the feature-based models, which are the most common ones used in information retrieval and personalization and the stereotype models.

- Interests, Preferences and Abilities

In recent years, interests and preferences play an important in the structures of user models. They are used as keyword vectors and hierarchical concepts by the interest-driven nature of the information. The research [12] used user preferences as a context. Abilities and disabilities, which consist of physical and mental also seem to be one of context to create a user model. We can find instances of using abilities in [13, 14]. For example, in [13] the mental ability of a user is investigated.

- Knowledge

  As time goes on, the user can learn new knowledge and forget some old knowledge, so the nature of knowledge is dynamic and will change over time. The changing nature of knowledge makes the models more complex than scalar models. Examples of knowledge models can be found in [15–19]. For example, the author in [16] used from a knowledge model to provide personalized services in adaptive hypermedia systems for adaptive presentation.

- Background

  The background information consists of any type of user background such as job, education, professional background, and also are stable over time. In most of the personalization systems, background information can be used as a critical component. Stereotype models are the simplest method to make background models as user models. An example of stereotype models can be found in [20].

- Goals

  Because of the changing nature of the goal models, goal recognition is difficult and not very accurate. In the goal models, the user purpose is modeled using a list of all purposes called the catalog. The research conducted in [20] used goals models in the catalog method.

- User context

  With the growth of adaptive and context-aware systems, the use of context models as the basis of user models has spread. For example, in [12], the author used a user context model to automatically adapt the system to the user needs. There are different contexts for users that a context-aware system tries to identify [21–23]. In [24], the author is categorized context in two groups. The first is the human user context and surrounding context and the second is according to persistence such as permanent (static) and temporary (dynamic).

- Relevant or non-relevant topics

  Although profiles are built from relevant topics but also irrelevant topics are investigated in [25, 26]. In these models, the system simultaneously identifies both kinds of relevant and non-relevant documents and if necessary, delete non-relevant information. Examples of these profiles can be used for filtering systems.

Depending on the problem, each of the discussed fields or any combination of these features can be used to build a user model.

2.2.2. Major Dimensions in User Models

Although there is no complete division on the user model's dimensions in previous works, we explored different design patterns for user models. In general, the user model can be classified in multiple dimensions, though often a mixture of them is used. Hence, we decided to present a comprehensive classification of different dimensions of the user based on research conducted in [21] where a three-dimensional space of user models is introduced. We have completed and classified user models to six dimensions as follows:

- Individual vs. group profiling,
- Supervised vs. unsupervised learning,
- Explicit vs. implicit feedback in user models,
- Long-term vs. short-term user models,

- Dynamic vs. static user models
- Distributive vs. non-distributive user models

In addition to the major dimensions defined above, there are also many sub-dimensions. There are different systems with significant differences that can follow from major dimensions, so the choice of the dimensions for the system is still a problem.

- Individual vs. group profiling

  An important aspect of user modeling is individual or group profiling. In individual profiling, the user information such as demographic information is considered while in the group profiling information of a similar group of users are gathered as a profile. The groups are formed by similar users with the same interests, goals, and preferences. There are a number of samples that are using user profiling in personalization as individual or group profiling.

  Group profiling can generate a partly comprehensive profile by overlapping a small amount of information between users. For example, in [27], is created a profile for a group of users‘ interests by user click-through data. This research used a learning method and a collection of training data to re-rank search results. Also, the author in [28] is used group profiling in two ways. In some cases, it is used of default values such as “woman” or “computer scientist” and in some cases, is used querying the user. In [12, 29], user modeling issues are discussed using stereotypes as a sample of group profiling and then used the group profiles to recommend new interests to users.

- Supervised vs. unsupervised learning

  There are two ways for producing profiles: top-down or supervised and down-top or unsupervised. In the supervised profiles also called deductive learning, the profiles can be created using the data mining process. This process begins with a hypothesis developed by a researcher and then the testing and validity process will be done. The critical problem is finding relations and correlations between hypotheses.

In the unsupervised profiles also called inductive learning, the patterns are mined and are entered the loop. Some statistical methods like classification techniques use unsupervised learning while other techniques such as rule-induction techniques use supervised learning.

- Feedback in user modeling

  For collecting information about user interests, we need to know about user feedback. The common used feedbacks in studies are explicit feedback, implicit feedback, and blind or ”pseudo” feedback. In explicit feedback, the information about a user to create his/her profile is collected explicitly by asking the user. In implicit feedback, user information can be implicitly collected on the client-side, such as browser agents or the server-side, such as search queries collected by search engines. Each method has advantages and disadvantages.

  Hybrid approaches are also possible. Hybrid approaches use explicit and implicit feedbacks on user profile creation. This approach combines the advantages of implicit and explicit feedback and increasing efficiency and the accuracy of information. Some papers [30] also proposed approaches for creating simulated feedback in personalization techniques and ranking functions. For example, Varma et al. [30] introduced techniques for simulating the user search behavior using click-through data.

- Explicit feedback or relevance feedback

  In explicit feedback, is asked from the user for his/her interests and preferences explicitly. This information is used as relevant documents to improve user profiles. The data collected from the user is in the form of Hypertext Markup Language(HTML) and the most contain personal information. For example, Nazar et al. [28] used questionnaires to collect information about the user in two fields of personal information management and adaptive visualization. The model has focused on static user characteristics.

  Explicit feedback has advantages and disadvantages. One of the most important advantages of explicit feedback is the correctness and accuracy of the information because the information is specified by the user. Some movie rating sites such as Netflix[1] provide the possibility of rating by users feedback or some sites such as MyYahoo![2], explicitly collect the user information and then use the information to organize the content of the web site. One of the most important disadvantages of explicit feedback is to cost the user's time and agreement. The issues related to privacy concerns is another disadvantage because sometimes users do not want to give some information. Since the user interests change, the profile also should be changed. So, we need a dynamic structure for the user profile.

One of the earliest studies on personalization based on explicit feedback is researched in [31] where authors have used explicit feedback for recommendation web pages. The user feedback is as a rating on links on a page. The authors have used ratings to recommend other similar links to the user. In [32] also a direct approach is used to identify a user's interest in web content to obtain explicit ratings on web pages from the user. The introduced system in [33, 34], also is used to assist users. This system used explicit feedback during browsing for help to the user. However, according to [35], providing explicit feedback from the user is not always desirable. Also, considering observations in [36], users are reluctant to provide explicit feedback. So, more studies tend to use implicit methods for inferring user interests.

- Implicit feedback

  There are several problems in explicit feedback such as the issues of data inaccuracy or data incorrect and also disruption to users. In implicit feedback, the user's behavior is extracted implicitly for creating a user model. The user behavior includes user clicks, dwell time, the information of scrolling, saving, or bookmarking a page. In research [37] implicit feedback is classified based on the resource to examination such as vote and click, retention such as bookmark and share, and reference feedback such as to reply or discussion. More useful methods for constructing the user model are user browsing history and clicked pages. In [38] authors are discussed the common implicit

[1] Netflix Website, Http://www.netflix.com/
[2] Yahoo Personalized Portal, http://my.yahoo.com/

feedback techniques. In the field of recommendation systems also exist applications of implicit feedback for collecting information from user behaviors. For example, [39–42] established a model by a user with monitoring click history and predicting future page visits.

- Short-term vs. long-term user models

  There are two important aspects in user profiling to infer user's purposes: collecting information about user interests, and the length of time which interests remain static or the period that the interests change.

  In short-term knowledge systems, the priority is to detect the variations for short periods or session boundaries and the profile just represents the user's current interests while the long-term systems consider long-term changes. The author in [43] used data collected over a long period of time in the query expansion technique. A system depends on different performance can use short-term and long-term data for different purposes for example in [44] author created user profiles as short-term and long-term using semantic concept hierarchy tree.

  Unfortunately, many existing models [45] has focused on the long-term user interests over time and have ignored short-term interests while users also have short interests and profiles have to adapt to the changes. There are a number of the ways for adaptations in [46–52]. In [49–51], authors incorporated information about the recent of an event and in [52] authors have used genetic algorithms into their models. The author in [48] benefited from a personalized system at different time periods.

  Other approaches to achieve adaptive in short-term interests would use retraining a predictive model periodically. Also, hybrid approaches can be used to combine several approaches of short-term and long-term interests. For example, consider a sportsman who wants to go on vacation, and he wants to reserve a hotel on the web. He searches for hotels. His user profile should save hotel information as short-term interests and he has several interests such as sport and so on as long-term interests. When the user returns from his vacation, the information related to hotels and reservation have to forget over time.

- Dynamic vs. static user models

  In static profiles, information remains unchanged over time whereas in dynamic profiles, as time goes on, the information can be modified or augmented. The earlier user profiles had a simple structure and static and there were not any learning algorithms. The model available in [50] focused on static profiling in the field of personalization.

  In dynamic profiles, the learning process always continues throughout the lifetime of the system. Because of the changing nature of user interests, the implicit feedback for creating the profile has easily adapted to dynamic profiling. The accuracy of using dynamic profiling is according to the time period. For example, in [53] the author has used a combination of static and dynamic profiling. The static profiling is collected by the online forms and the dynamic profiling is learned and analyzed by the user's data.

  Since the nature of online systems is dynamic, they have to use dynamic profiling. For example, in [54] is started to discuss sudden changes in interest in contrast to gradual. In [55] also is presented a new technique based on user interests' shift. The model is developed by the Bayesian approach for filtering systems and different experiments for the evaluation of interest shifts in the research are conducted. In [56] user profile is adapted itself using new information collected.

- Distributive vs. non-distributive user models

  Group profiles can be divided into distributive (decentralized) or non-distributive (centralized) profiles. In distributive profiles, the properties are distributed to all components of the body as same, while in a non-distributive profile they are not distributed [57]. Since the components in the non-distributive profiles communicate with the system using a client-server model, They have several advantages such as stability, more security, and consistency. They are also efficient because does not need redundant storage. The non-distributive profiles have also disadvantages such as providing network connections and hardware resources.

Nowadays the non-distributive systems are more practical, because of how collecting and information from various clients. This collected information can be applied by personalization systems. There is an overview of distributive modeling in [58]. In [59] the author presented a general user model ontology for distributed user models. In the paper, the user's information such as age, position, birthplace, and also user's preferences are collected. The advantage of the distributive profiles is the simplification for exchanging user model data between different systems. In other words, in the presented model in [60], the structural differences between user modeling systems are removed by a specialized ontology for user modeling tasks.

2.3. User Model Structure

How to collect information about a user, how to represent information, and generally, user model structure is different in each application. Besides, the used techniques to create and represent the user model is also different. Concerning the representation models [61–64], user interests can be represented as a set of rules or keyword(term) or concepts [31, 65] such as vector space or class vectors [66], graph-based representations, an instance of predefined ontology or a hierarchically-arranged collection of concepts [39, 67–73]. There are several studies on the user model representation techniques in [74].

After storing the appropriate data in the user model, it is necessary to use an effective representation model to structure the user profile. More recent approaches consider concepts and relationships between concepts and requiring an external knowledge resource, such as the ODP[3], or Wordnet[4].

2.3.1. Keyword based Models

The main feature of keyword-based profiles is simplicity and flexibility. They use a set of weighted words to represent. These words are extracted from the web pages. Although

[3]Open Directory Project, http://www.dmoz.org
[4]http://wordnet.princeton.edu/

there are many relationships between sets of keywords, keyword-based profiles cannot model user interests precisely and correctly. However, to employ keyword-based models for web personalization, we need to acquire keywords from documents visited by the users. For example, in [75] is proposed a combining approach to build a keyword-based profile on extracted keywords and the user's explicit rating of the page.

In [76], the author presented an approach to creating a keyword-based user model where keywords are extracted from the visited web pages. To combine different approaches to keyword extraction, the paper developed an extensible library that focused on the extraction of required keywords from web pages. So to extract relevant keywords from web pages, it needs to employ and combine other techniques like semantic knowledge. There are problems like the vocabulary problem [77] and the conceptualization problem. In the conceptualization problem, there is a difference between user concepts and available concepts.

There are several modified versions of keyword-based user models. For example, in [58] the author tried to solve the lack of semantic information of keyword using a predefined knowledge base. In this way, the user model is improved by combining the keywords and ontology concepts.

2.3.2. Vector Space Models

The vector space models are first introduced in information retrieval [78] and then are used for representing the documents [79] in the user profiling field. They are represented in the form of vectors of weighted terms. In [56] the authors have developed a user model based on the vector space model to create user profiles using browsed pages and bookmarks.

In the vector space model, concepts are trained on examples by an external pre-classified database such as an open directory. Then there is a need for mapping between vocabulary and concepts. Finding and handling an open directory could be considered as the most problem in this method. In the vector space model, the first pre-processing is performed on the interest documents for removing interjections and particles. Then the feature selection phase selects key terms and their weights are calculated by different methods. Finally, the weighted vectors and updated time is returned to the user profile.

2.3.3. Semantic Network Models

At first keyword-based user profiles were used to represent the user's preferences. But as keywords made a lot of ambiguity such as polysemy problem and user preferences could not be represented by a list of keywords, thus semantic approaches were created. Without semantic structure, content processing is not possible. There are complexity factors in natural languages such as synonyms, polysemy. Semantic approaches can provide the semantics of user preferences in a conceptual way [80–82].

Semantic networks are made up of nodes and arcs that the nodes represent concepts, and the arcs connect the co-occurrences nodes. Semantic networks need to analyze a large amount of user data. In [83], the author has described a personalized search by a semantic user profile to connect related concepts. [84] also presented a five-step process based on term dependencies and as a hierarchical network.

2.3.4. Concept Hierarchies Models

The concept Profile is a semantic network-based profile that the nodes represent abstract topics instead of words. In spite of, the more systems in the past works, was structured information as keyword vectors like the conducted works in [16, 85–87] but there are systems to store profiles as concept hierarchies such as SmartPush [88] that was consist of 40-600 nodes.

Hierarchical concept profiles in the simplest form are built by a reference taxonomy such as WordNet or thesaurus and in the complex form are built by reference ontology such as ODP. There are machine learning techniques such as classification techniques and clustering algorithms to build hierarchical user interests. Although generally, the hierarchical algorithms use the reference taxonomy to create user profiles, they can be created without any reference taxonomy like [89] that the author has provided a hierarchical user model as manual. In research [90] also is provided a hierarchical clustering by unsupervised methods. In [89] also profile is clustered as a concept hierarchy that bottom and nodes are short-term and the upper nodes are long-term user interests. Yihong et al. [91] presented a method with two steps to create user interest. This method the first cluster the viewed web pages from the HowNet[5] and then in the second step, calculates the similarity between the keywords and the vectors.

Although hierarchical clustering algorithms are accurate, classification methods as standard are more reliable than them. The classification techniques require a knowledge source, therefore, collect web pages for training data in each taxonomy.

Several different text classification techniques have been used such as SVM[6], KNN[7], Naive Bayesian, Decision Tree, and Neural Networks. For example authors in [92] have employed text classification for building Chinese weblogger's interest. They have used the combination of classifiers such as the Naive Bayes Classifier [93], Support Vector Machine [94] and Rocchio Classifier [95].

Authors in [96] created a user profile using a three-level hierarchical structure. Similar work in [96], Trajkova [73] used a classifier to construct user interests by user's browsing history in a three levels taxonomy. The model in [97] also has used the ODP taxonomy but the difference is that it has used the user search histories to build his profile. One of the most limitations of classification methods is when the taxonomy changes. By changing the taxonomy structure, the training data must also change.

In [63], the researcher made the user and general profiles using a set of categories by ODP based on the user search history. The purpose of references [98] is to show that a user's interest focus on more than one domain at the most time, and so it can use hierarchical structure in works such as recommendation, personalization and information retrieval. In [99] is constructed a user profile by architecture tree concerning Google directory. In the paper, each topic is presented in a tree directory. In [44] is also created a user profile with the

[5] Available in http://www.keenage.com/zhiwang/e_zhiwang.html
[6] Support Vector Machine
[7] K-Nearest Neighbour

semantic concept hierarchy tree to solve the disadvantages of a lack of semantic information of keyword. A navigation based system for browsing web sites has proposed in [67]. The system has used a hierarchy user profile.

2.3.5. Ontological Models

An ontology defines the existing concepts and relationships between users. The aim of using ontology is to make relationships between concepts. The start of using ontology to representation user model was in 2005 with [100].

As we mentioned before, a user profile can be created by a pre-existing reference domain ontology [101]. The ontology can be created manually collecting user's data to model the user information [50] or automatically for various domains [102]. For example, in [103], is utilized learning techniques to find relevant classes in taxonomy and relationships between classes. There are many surveys of methodologies in [104–106].

The advantage of ontological models is that there is no burden of designing a new schema for a particular application. One of the most popular reference ontology is available in RDF format in [50]. The research [22, 107, 108] has used ontologies for knowledge representation in various applications such as browsing, query extension, and personalized retrieval. Also, using an ontology in the field of user profile has provided in several applications [50, 73, 109] personalized web search and browsing [65, 67].

In [73] is created a keyword-based user profile by a pre-existing ontology. In the paper, it is investigated techniques that implicitly build ontology-based user profiles by the user information. [110] has built an ontological profile with the user's dynamic information. In the field of personalized web search, [67] has generated a set of ontology categories by mapping the content of each page to a group of categories in Magellan ontology. Liu et al. [65] have also created a user profile from pre-classified documents with mapping a user's search query to ODP categories. They have also classified any source of representative text by the system. Nazim et al. [111] studied domain ontology as reference ontology using ODP.

User documents are extracted in topics or concepts and a feature vector is built for each topic for creating the initial user profile.

Using ontologies in the different applications, there is a need to create a general ontology for modeling user profiles where this general ontology can be employed as a reference when profiling functionalities need to be developed. [50] has presented such a general ontology for modeling user profiles for different applications and communicating between them.

On the other hand, ontological approaches are applied in addressing the unseen data or the cold-start problem [112]. For unseen data or unavailable user behavior, existing concepts in the domain ontology can be used as a user profile [73].

2.3.6. Topic-based Models

Topic-based user models extract topics of documents by using different methods. One of the common methods to extract topics is using a reference ontology or ODP. In some research, ODP is also used as a reference to topics extracted from trained web pages. For example in [113] the user interests are used as a classification on ODP metadata to personalize search and [114] used ODP for classifying topics and explored on their relations with query ambiguity. Although this method has advantages like extracting the hidden topics in documents, the sparse topics are a problem in extracting topics in this method. In addition, training pages with labeled topics is also a big problem.

There are several latent topic techniques to find text content features such as probabilistic latent semantic analysis (pLSA) and LDA. Topic models discover and analyze the basic semantic structure of a set of documents. Then the hidden structure is modeled using Bayesian inference on the topics that are supposed to be in the collection.

Probabilistic topic models are also used for personalization. They use Probabilistic Latent Semantic Indexing (PLSI)[115] and Kullback-Leibler Divergence to estimate a query model.

Hofmann [116] proposed pLSA that uses a statistical model to extract the topics from the hidden semantic structure of documents. PLSA makes improvements in probabilistic modeling of text documents, but have the overfitting problem for the unseen document.

LDA was proposed by Blei et al. [117] because of the problems of pLSA. The LDA is an unsupervised model that overcomes the problems of overfitting in the pLSA. In [118], the authors report the result of comparing between the LDA with the multinomial mixture(MM) and the pLSA models. In this thesis, the LDA is used in various parts of the research to measure the document similarity based on latent topic distributions.

Recently some research tries to combine topic distribution models with features of query ambiguity for improvement in personalized search. There is some research on the improvement of the personalization process for some queries [54, 55, 119]. In [43], the authors discussed the relation between query ambiguity and topic distributions. In [120] a new approach is proposed for the comparison of click entropy and topic entropy for identifying ambiguous queries.

2.4. Learning Techniques for Creating a User Profile

After the organization of user information, it needs to define user profiles by a standard structure. Whatever the chosen language was more structured and expressive, the user model would be accurate. There are a number of techniques with different characteristics and performances for constructing user profiles. The early traditional approach for user modeling was knowledge bases and there were problems with these knowledge bases: the construction was a resource-intensive process namely obtaining such knowledge bases was a problem, and after construction usually, they are not extendable. In addition, with the increasing of data, these knowledge base problems and uncertainty in user modeling were increasing. The need to address these problems caused to rise of statistical models for user modeling.

After the traditional user models, they were constructed using different learning techniques such as the probabilistic models, linear models, TFIDF based models, Markov models, vector space model, neural networks, decision tree models, rule base models, genetic algorithms, Bayesian networks, classification and clustering methods. We argue these techniques in the following.

2.4.1. Classification Techniques

Classification methods are unsupervised learning and cluster the objects according to their attribute values in the same group. Liang in [17] utilized the machine learning techniques consist of Rocchio, kNN, and SVM methods. This paper also measured the effect of the increase in the number of training documents on personalized search performance. In [61] also the text classifier is used for personalization web pages. In this paper, websites are classified into regions as hierarchical.

Michael et al. [113] argued a profile that was implemented as classifiers for new documents. In this research, a comparison of several different classifiers is done, including Decision tree, Nearest neighbor, Neural networks, and Bayesian methods.

2.4.2. Predictive Statistical Techniques

The predictive statistical models for the first time were used to model the user using the machine learning techniques [121]. There are a variety of uncertainty techniques in the fields of machine learning called predictive statistical models. Two main approaches are content-based and collaborative. Although both the content-based and collaborative approaches are aiming to estimate the user behavior, in the content-based approach it is done by the user's history while the collaborative approach uses the behavior of similar users.

Many statistical models are used in the content-based and collaborative approaches. In [122, 123] is conducted comparative studies on content-based predictive models. [122] has used two rule induction techniques for learning and predicting the result obtained by a student when performing subtraction. Davison et al. [123] made a comparison to estimate the user behavior using the decision tree and the Markov model.

Macskassy et al.[124] made a comparison of a naive Bayes classifier and a rule-based model in recommendation systems. The results indicated that Bayesian networks are more accurate in different ranges and conditions. Alspector et al. [125] investigated a comparison between the collaborative approach with a content-based approach in the domain of film recommendations. According to results, the collaborative approach using linear networks was more accurate than the content-based approach using decision trees. They concluded that each approach has limitations and the best results are obtained when both the content-based and the collaborative approaches are combined.

2.4.3. Neural Network Techniques

Neural network methods were started in 1988 by Gardner et al. [126]. This method has a non-linear network structure. In content-based approaches are used of neural networks to represent a user's preferences. Jenning et al. [127] used a neural network to create a user profile where the nodes in the network are liked terms by the user and the edges are an association between terms.

In another research [33, 34] authors used user browsing behavior to construct neural networks. In [24], the author also used nonlinear approaches to create user models by collecting user behavior from implicit feedback. In the paper, the profile is based on a multi-layer neural network method where the feedback is reading or rejecting a document.

2.4.4. Bayesian Network Techniques

Bayesian networks have been used for a variety of user modeling applications [128, 129]. The structure of Bayesian networks is based on Bayes theorem and conditional probability distribution as directed acyclic graphs with nodes and arcs. Bayesian networks are more extensive in comparison to other methods discussed because they can represent any probability distribution instead of a single variable. For example, [130, 131] used a Bayesian network to estimate agents' behavior.

There are several studies in the application of Bayesian methods to the task of modeling the user interests. In [20] the authors designed a Bayesian model to infer a user's behavior by the user's background, actions, and queries in detail. [132] has used a Bayesian network to model the next query search of previous query behaviors on the web. The other model in [133] applied a naive Bayesian classifier that calculates the relation between an item and a specific class.

On the other way, Bayesian networks are a practical and effective technique for modeling user interest shift and tracking for example, [55] has used a Bayesian network to create a profile. In this paper, an interest tracker is employed for learning based on relevance feedback. In [134] a new user modeling system called Zebra is proposed and provides an inference mechanism for reasoning out new information about students. In this paper, a Triangular Learner Model has proposed that used the Bayesian network and Markov model. The core of Zebra is consist of two engines: a mining engine and a belief network engine.

2.4.5. Linear Models Techniques

Linear models have used weighted sums of known values to make unknown values. They have a simple structure and easily extended and generalized. Collaborative and content-based approaches use linear models for different applications as predicting a user's rating for news articles. The resulting is the weighted sum of the ratings.

2.4.6. Rule Induction Techniques

These techniques use the set of rules for prediction. They can represent rules in a variety of forms such as decision trees or conditional probabilities. They are used by both the content-based and collaborative approaches. For example, in [135], has used a rule induction based system to predict a user's next action. In [51] also is combined a C4.5 technique with a rule induction technique for tutoring applications.

[136] used a combination of rule-induction, TFIDF-based and linear models. In this paper, authors have used two models to predicate and recommend news articles. When we don't have enough data and information in the system, for building an initial model this technique can be useful.

There are several studies under the collaborative approaches that use a ripper to learn a set of rules for different tasks like [137–139]. For example, [137] used rule induction techniques to predict ratings in films. In [139] the authors used ID3 to learn a decision tree to predict user next action on a scheduling problem.

2.4.7. Markov Models Techniques

Markov models have a simple structure based on the Markov assumption. Using the Markov assumption, the user's requested page can be predicted by the last visited pages. Some researchers have used the Markov model to predict users' requests on the web. For example, several models like Bestavros' model and Horvitz's model considered and calculated the probability of request a specific document and then used these generated predictions by the systems to help users.

3. LITERATURE REVIEW

In this chapter, we review related works based on the conducted research. This chapter aims to three main areas: the need for personalization, personalized search, and user profiling. The user model has been investigated in several aspects including the user model structure and representation methods. According to user model structure and user needs, there are many possible applications on personalized systems. We define a general explanation of the need for personalization and especially search personalization. Then we explore the approaches to construct a user profile. Finally, several applications on ranking approaches using user profiles conclude this chapter.

3.1. The Need for Personalization

Since in the search process, queries are short [140, 141] and the ranking process is a problem for identifying relevant documents, the need for personalization feels in recent years. Teevan et al. [39] have investigated people's aims on the same queries and reported some results. Based on the results, the need for personalization is different in queries and different people.

The most recent works have conducted on personalization using user interest profiles for all queries [120, 142–144]. However, based on the research conducted in [120], some queries can benefit from personalization. For some queries, personalization does not make an improvement that it is maybe due to the effect of aggregating group information. Teevan et al. [7] argued that personalization only for unambiguous queries can make improvements in the ranking process, and [6] also discussed the concept of potential for personalization and completed her previous work on selecting between personalized and un-personalized rankings.

There is a need to estimate the potential for personalization for queries. In order, this exploring the characteristics of the query is necessary to achieve a correct personalization process. Due to the importance of the query in the personalized search process, more recent works in personalization has conducted on the potential of the query. In some research, only query features are considered such as the length of queries or query click entropy [140] while in the other research the clicked documents and dwell time is explored. Based on the reference [145], personalization is insufficient with low click entropy queries. In [143] also in a similar method, is proposed user entropy which averages click entropy by each user and discussed that the user entropy is useful for low-frequency queries.

3.2. User Profile Representation

There are different ways for creating and representation user profiles such as weighted vectors of the original terms (term-based methods) [141, 146], topic-based [74, 144, 147] and ontology-based methods [27, 39, 60], however the main focus in creating user models in personalized process is the keyword-based user profiles.

Gauch et al. [67] proposed a keyword-based(term based) user profile by the user's search histories to improve the personalization. Weighted keyword vectors are used as a predominant representation of user preferences [119]. Trajkova et al. [73] created a keyword-based user profile via a proxy server. In the paper, user information such as clicked document and dwell time are collected by implicit techniques to build a user profile.

McGowan et al. [56] created the user profiles as a collection of vectors for different topics. Although this method has advantage, one of the problem is the relation beetween the independence topics. To solve this problem, [148] provided to use a concept hierarchy between topics. So the topic-based methods can be used to overcome the problems of the keyword-based approaches in the personalised search [4, 149–152]. In topic-based methods, the user's profiles are presented as a probability distribution over topics [4, 149, 150]. In this approach to create user profiles, the topics are identified from user documents and then are extracted using a knowledge base [149, 150], or some techniques [4, 151].

The other common method to create a user profile to improve personalized web search and browsing are ontologies [153, 154]. [84, 155, 156] ontology-based methods are applied using the ODP taxonomy as the web topic ontology [37, 157]. [38, 134] used the ODP knowledge base for creating user profiles in the field of query expansion. Michael et al. [113] proposed the user interest as a classification on ODP metadata to personalize search and [114] used ODP for classifying topics and explored on their relations with query ambiguity. Eduardo Vicente et al. [158] conducted a study on different user profile representation approaches.

In the field of ontological user profiles, there is some popular knowledge base such as ODP concept hierarchies or Wikipedia. The ODP[8] is used widely as the most popular directory produced by humans in personalization [149, 150, 152, 159]. Two levels of ODP is depicted in Figure 3.1. and the extracted topics are listed in Figure 3.2. The two figures are taken from [160]. The ODP has the problem of not having access to the proper classification for some documents. Therefore, it is necessary to create topics automatically. Topic modeling techniques can overcome the limitations in the ODP [117] using Latent Dirichlet Allocation [4] and other techniques to extract topics from documents.

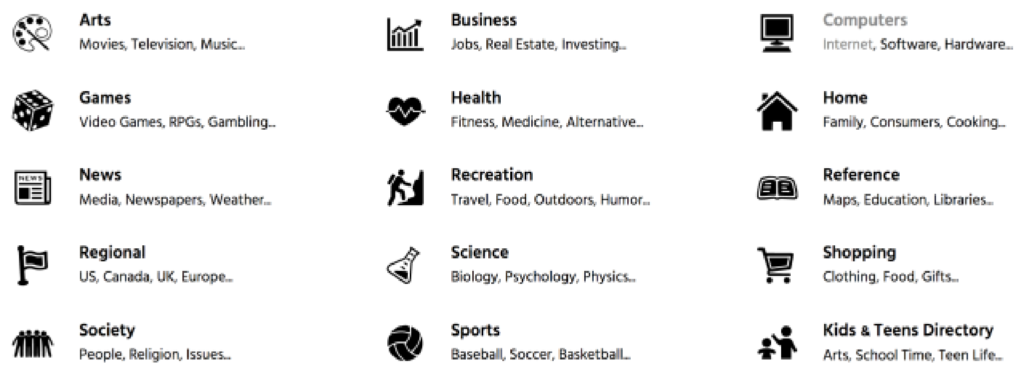

Figure 3.1. ODP with two levels of categories.

Some papers preferred to use the structured data instead of keyword-based profile such as SiteIF [87] that authors utilized semantic networks or Anatagonomy [161], Letizia [162, 163], Krakatoa [164], Personal WebWatcher [165], and WBI [166] that search behavior is used to create profile. In a similar way [17] described a personalized search approach using a semantic graph structure from ontology. User profiles are built based on queries in a search session. The tracking changes mechanism is defined using the Kendall rank correlation measure.

[8]http://www.dmoztools.net/docs/en/about.html

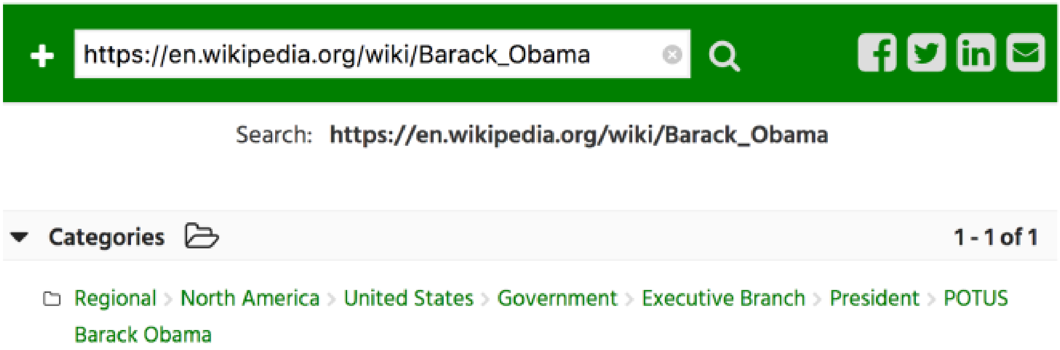

Figure 3.2. An example of topics extracted from ODP.

Hybrid methods of keywords and concepts are also used by [68] where the authors tried to combine the keywords and the ontology concepts. The purpose of this structure is to solve the disadvantage of the lack of semantic information of keyword. Sieg et al. [167] presented an approach that uses a spreading activation to build models of user context as ontological profiles. In the paper, interest scores of concepts represent the user's interest. James et al. [168] proposed a model for a personalization search that captures the user interests as separate ontology-based concepts using clickthrough data. Eduardo Vicente et al. [158], conducted a study on different user profile representation approaches. Amer et al. [169] investigated word embedding in the field of personalization and did not lead to good results because of user profile content and structure.

There are efforts to provide the generic user profiles among the different research such as Golemati et al. [50] tried to develop an ontology as a reference model to create the user profile. Razmerita et al. [170] also presented a generic user model architecture, and Liu et al. [65] created a comprehensive user profile by combining user's search behavior and the ontology concepts.

3.3. Personalization and Topic Models

A newer discussion on the topic model and personalization is by Harvey and Carman et al. [4] and [5] that the authors introduce new models based on latent Dirichlet allocation for personalized search. Thanh et al. [171] used latent topics for modeling search tasks by an unsupervised topic modeling method. A learning method was used to train ranking models. Than et al. [172] also introduced an embedding method to learn user profiles where users were embedded in a topical interest space.

In some research [173] probabilistic topic models are used to improve the personalized search such as [173] that a probabilistic topic model was used as user interests. In this paper, the Kullback-Leibler Divergence was used to estimate the query model. Shao et al. [174] proposed a text similarity algorithm using Jensen- Shannon distance to model text set. This paper used the topic model and word co-occurrence analysis to calculate topics in the text. Song et al. [175] provided the methods for adopting a user-specific RankNet model using neural networks. These methods were evaluated based on KL-divergence and click entropy.

Nguyen in two research [176, 177] proposed a three-way method based on the user query or clicked documents by the three vectors and then used in a deep learning architecture to rank documents. Momtazi et al. [178]employed topic model in the field of query suggestion. In this query suggestion process, the topic model was used to extract semantics on the AOL query log. The main problem in this paper is unseen queries or queries without a history in the searching process.

3.4. Personalization

Personalization emerged to overcome overloading data on the web. Whatever the user profile created be more accurate, the process of personalization is more developed. Personalization is applied by personalized services, such as personalized search, recommendation services, filtering systems, browsing, and navigation systems [65, 179]. For example, the authors [86, 180–185], are focused on the personalized filtering using user's search histories in Email, Electronic newspapers, Usenet groups, and Web documents.

Today more research focuses on search due to the importance of search as one of the most common activities on personalized web search [87, 186–189]. Commercial products are also provided personalized services like Google Lab's personalized search[9] which matches results with user interests that are provided by themselves explicitly. Pretschner [190, 191] provided an overview of the personalizes services by describing approximately 45 personalization systems.

3.4.1. Personalized Web Search

In many search processes, the user information can be searched using a query and as a list of results. But these results cannot be always enough and there is a need to search systems to personalize results. There is a need to create user models to increment the effectiveness of the search outcome such as [192, 193] that try to combine an exploratory search approach with an adaptive visualization for increasing effectiveness in personalized search.

There is a lot of solutions for personalized web search such as profile-based approaches and click-based approaches. The profile-based approaches use user interest models and are very efficient while the click-based methods are based on clicked pages according to the user's query history. Profile-based methods use some techniques for personalized search such as browsing history, query history, bookmark, saving, print, and so on. For example, the authors [87, 161–163, 165, 166, 185] increased the efficiency in personalized search and navigation support with using browsing behavior. Qiu et al. [46] described a personalized search system to predict future queries and Lai in two separate works [194, 195] tried to combine the user searching profiles and the document profile, for presenting customized search results to the users.

In the web search context, Dupret et al. [196] constructed a user profile from user interaction logs aiming to make web search engine evaluation. Smyth [197] proposed a collaborative web search approach as a community-based approach that reflects and shares knowledge within search communities.

[9] Google Personalized Search, https://www.google.com/psearch/

The use of ontologies for creating user profile also cause to increase the efficiency of personalized search such as [198] that proposed a new web search personalization approach in the form of concepts by mining search results or Nazim [80] proposed an ontological user model that used WordNet[10] as a knowledge base. The authors [119, 145, 199] have addressed the main issues related to privacy for information retrieval and proposed that the user profiling process should be done in the client-side because of privacy issues.

There are two major types in personalized search: re-ranking search results and modifying the user's query or query expansion. We would argue about them in the following.

Re-ranking search results

In re-ranking systems, the created user profile by user feedbacks helps to create a new ranking based on ranking algorithms such as [46, 47] that a user profile is created based on short and long-term interests.

Re-Ranking algorithms [149, 200] have used a ranking function [155] to sort the results of search engines. For re-ranking results, a spreading activation algorithm is used. Oard et al. [40] used the Personalized PageRank approach using a user profile to estimate the ranks of the results. In a similar work Aktas [201] introduced the re-ranking web search based on the PageRank approach using the user previous searches for making hub scores.

Pretschner [61] described a metasearch engine that learned from visited pages and re-ranks documents returned based on the profile. Daoud et al. [83] exploited a semantic user model of related queries where the user profile has a graph-based and session term structure.

Query expansion

Query refinement is to improve the initial query using implicit or explicit information. This technique was discussed by Salton and McGill in 1983 and then was used to solve the word mismatch problem in the words used by search engine [202]. The purpose of this process is to obtain the semantic or hidden concepts behind the normal keywords or queries by external

[10] http://wordnet.princeton.edu/

sources like background knowledge. The query refinement could be also done by the user model to improve search effectiveness however since it needs the relevant information, it is a difficult job.

Finkelstein [203] proposed a system that generates additional query terms based on the text surrounding the original query terms in a document the user is reading and [204] proposed an automatic query expansion approach using an experiment to compare the performance of different inputs in searching and browsing such as search query, expanded query, snippet, and page content. The results showed the best input for identifying user interests was the expanded query.

Another issue that results in emerged query expansion is short queries with a length of about two [205]. The authors [65, 203, 204] investigated the query length using co-occurrence concept and identifying the most frequent word in the field of query expansion.

In the field of personalized search, [153] used user profiles to expand queries and [206] refined queries using a local context. [207] also refined queries only based on user preferences and without user profiles. Kim et al. [208] used a fuzzy concept and query expansion in the personalized search engine by using link-based search techniques. Many search engines used link structure and fuzzy concepts with a network to find user queries ' subjective interest and their results were efficient. Nazim [111] exploited a query generator by the user profile ontology for generating a personalized semantic search.

3.4.2. Personalization in Various Domains and Applications

We discuss several applications and their impacts. The common feature of all applications in various domains is that user modeling is utilized to provide personalized information from the web. We intend to categorize personalized approaches based on the proposed categories [189] as follows:

- Approaches to improve ranking (re-ranking) search results

- Approaches to modify the query (query expansion)
- Approaches to recommend items (news, film, paper, and . . .)
- Approaches to filter irrelevant information

The approaches in the first category propose modifying the ranking score in the PageRank algorithm to re-rank results by user profile such as [68, 114, 145, 209, 210]. The approaches in the second category are aimed at expansion query by user profile as knowledge support. These approaches want to obtain more accurate queries by using user profile and query expansion techniques such as [47, 65, 211, 212]. The third category approaches try to suggest items and web pages of user interest that are known as recommendation systems [213].

In the previous chapter, we discussed two first categories and in the following, we consider the other applications on personalization such as recommendation, filtering systems, and then we review the other applications that the user model is the main component in them.

Personalized recommendation

Nowadays one of the most important fields in the personalization on the web is recommendation systems. Recommender systems are applied in web sites such as ”Amazon”, ”Netflix”, social networks, websites, and news. There is some research on semantic web personalization specifically recommendation systems [22, 214–216].

Recommender System collects the requirements of every user and after doing a process, recommends user-specific items to them. Recommender systems can use user profiling to collect and process user requirements. In the field of recommendation based on the user profile [154, 155] provided a recommendation based system that covers the changes in user interests extracted from the user documents.

There are several methods like explicit and implicit feedbacks to collect and select items for recommendations such as clicking items, bookmarked items, rated, buying a product, requesting a movie, recently viewed, etc. For example, Liebermann and Budzik [162, 206] presented a recommender system to suggest items to users. In these papers, the user information is collected implicitly from the recently viewed web pages.

Ouanaim et al. [217] presented an application to recommend the closest tourist spots to the user using his/her information. The system provided in [218] called Case-Based Profiling for Electronic Recruitment(CASPER) automatically recommends jobs to a user based on his qualifications and experiences.

Zhao et al. [219] created a personalized recommendation system using the behavioral factorization technique on Google plus. Ryen et al. [220] presented five variant information resource including social and historic resource to predict future behavior.

The recommendation systems can be also a type of information filtering system that filters irrelevant information and recommend relevant information to users such as [221] that presented a recommender mechanism called StumbleUpon and uses collaborative filtering while requires explicit action from a large community of users. These social networks need to make virtual communities for the distribution of web content.

Personalization in Filtering Systems

Filtering systems using different techniques try to decide which documents are irrelevant and which are not and then delete some irrelevant documents. This is usually done by ranking functions with a specified threshold. Since the decision about relevant or irrelevant is difficult, so the system performance in ranking usually is better than filtering.

There is some research on filtering techniques [140, 168, 222–226] and intelligent agent [19, 162, 227–229] that we discuss in more details below. Hyunjang et al. [102] used a filtering system for personalized documents on the web. Lam et al. [55] investigated a filtering system using a Bayesian approach to track changes in interests.

Content based filtering

In content-based or cognitive filtering, the user profile is created using exploring the content of visited items by the user and feedback is very important. In this technique, the main goal is to find users with similar explicit feedbacks for the items.

Collaborative filtering

In collaborative filtering, the users are considered in the groups and are clustered with similar interests. The clustering in this technique is very important and has a great impact on system performance. There is another method for filtering information like demographic filtering systems or fuzzy clustering systems [230]. In demographic filtering systems [231], demographic information is used to recognize user preferences. One of the advantages of collaborative techniques is to support diverse ranges of content, such as image, video, music as independence from the content.

Foltz and Palleti [140, 232] is developed a filtering system that user profiles were created using both implicit and explicit user feedback. Chen et al. [153] proposed an approach to collaborative web search using the search behavior.

Personalized summaries

From each document can be extracted summaries that show the main content of the document. There are several techniques for the selection and extraction of a summary. The early research on summarization [233, 234] showed that users tend to select the parts of the text that are related to their interests. Having a user model, the summaries could be personalized for example for a digital newspaper, an automatic summarization process can help to personalization. The presented system in [235] sends a selection of the news items to the users concerning their user profiles. Tombros and Carbonell [236] and [237] generated the summaries by combining different fields such as position and title.

Other applications

Several aspects of user modeling are used in different applications such as personal information management, energy management, adaptive hypermedia, adaptive visualization, intelligent tutors, electronic books, intelligent agents, adapting to physical limitations. Orwant et al. [238] presented a user modeling system called DOPPELGANGER for filtering and sorting electronic mail. In the system, users' information is gathered and then processed. DOPPELGANGER used sensors, applications, and learning techniques like linear prediction and Markov models to model users in a multi-sensor, multi-application environment. Each user model is a dimension that is determined by the available sensors such as the user's political preference or her/his age, favorite color, and so on.

In the field of personal information management and adaptive visualization, the model, presented in [50], focused more on static user characteristics. In this paper, to complement the user profile ontology, a combination of explicit and implicit feedback is investigated. Using user profiling in the field of energy management is also investigated [239]. In the paper, energy management based on user profile and micro accounting for smart energy management is developed.

Adaptive hypermedia systems tried to present relevant information base on user-specific characteristics, goals, interests, knowledge, and abilities. In intelligent tutoring systems, it is tried to identify user needs and help to students using a user model. Using the user model, the system can adapt appropriate exercises and examples to this user. Intelligent tutors use learning techniques to adopt knowledge and so need to possess a user model.

Early user models in intelligent tutor's systems were very simple and efficient. For example, Carroll et al. [240] built a training system that can not adapt to user needs. Then later systems tried to determine what the user knows. They started to build a student model and tried to add adaption property to the learning system based on the current student model. For example, Federico [241] used user interests and weaknesses in finding solutions to problems. Anderson et al. [242] presented a model-tracing method that knowledge is represented in terms of productions. In another sample, [243] used a Bayesian model to help students.

User models are also applied to facilitate interpersonal information acquisition and retrieval such as Harvey and Bull [244, 245] that used user model based on preferences expressed by the user and user actions to facilitate information retrieval. The authors in [246] described intelligent information agents and the key limiting factor in agent technology. This paper has discussed building and maintaining ontologies as a body of knowledge for the web as an important factor. Ontologies can also classify or linking items to create communication between user and computer. Trewin et al. [247] have utilized user modeling to dynamically adapt to physical limitations. This paper has used keyboard operations for personalized user modeling with various disabilities.

4. ESTIMATING POTENTIAL FOR PERSONALIZATION

In the process of personalized search, personalization is not appropriate for all user queries and may even yield worse results than generic ranking methods. The ranking for a navigational, specific, and unambiguous query is usually stable and its ranking does not depend on the user preferences. Better rankings can be obtained for those queries without personalization. For example, the query “myspace” is usually a navigational query for the social networking website regardless of the user issuing this query. For such a query, trying to personalize can produce an inferior ranking.

In this chapter, we consider the potential for personalization in two phases. In the first phase, the query features are explored to identify effective factors to measure personalization in queries in subchapter 4.1. In the second phase, the potential for personalization for queries is estimated. Finally, in 4.3. we quantify the query personalization using experiments and provide the results.

4.1. Identifying Effective Factors to Measure Query Personalization

We investigate the appropriate and effective factors to estimate query personalization. In order, we explore query features in several aspects including query structure, query history, and the clicked results for each query. Then we explore which features are the most valuable to predict query personalization to help personalization services using prevention useless personalization.

In this thesis clicked results by the user are used as the main resource to research. As we know, people clicked on the same results for common and popular queries such as ”Google” or ”Facebook” but different results for ”learning PowerPoint” or ”morning exercise”, so the entropy of clicked results for each query is also important. Based on the conducted research by Teevan et al. [6, 7], there is a direct relation between the potential for personalization and click entropy. The results obtained in the research show a correlation around 85% between personalization and click entropy. Like the work done in [7], click entropy is calculated based on Equation 1 which measures the variability in clicked results over people. When more pages are clicked for a query, click entropy is high and conversely, a small click entropy shows a few pages are clicked for a query.

On the other side, query frequency is also an effective factor to measure the potential for personalization in queries. The query frequency can be collected from the query log. It is assumed that high-frequency queries are the indicators of the popular and navigational queries and are not appropriate for personalization and conversely low-frequency queries are appropriate for personalization. Here our goal is to investigate the relationship between click entropy, query frequency, and personalization.

To illustrate the relationship between query frequency and click entropy, we depicted Figures 4.1. and 4.2. for the queries in the AOL data set. Here, we can deduce from the figures a relationship between query frequency and click entropy that with the increasing the query frequency, click entropy is also increasing. But as the graph shows, there are some irregularities and it is not a strictly increasing graph. For queries with low query frequency (in our data set less than one hundred), the graph is ascending.

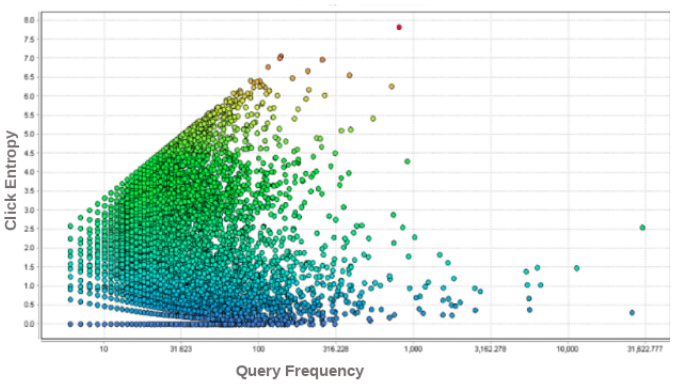

Figure 4.1. The relationship between query frequency and click entropy in the AOL data set.

According to the above figures, with increasing the query frequency, click entropy also increases but after reaching a certain extent, click entropy does not increase and remains relatively constant or even in some cases decreases. Figure 4.1. illustrates this issue well. This

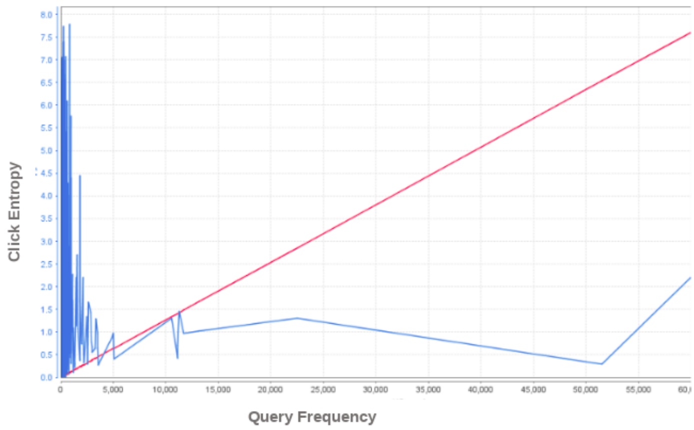

Figure 4.2. Rate of changes in click entropy to frequency

stable mode is somewhat related to the popularity of queries. Popular queries are known queries with the high frequency used by people such as the names of country, city, the special dates and the names of models, book or application (Microsoft office 2006), the names of movie, music, the names of popular persons (sportsman, actors, political persons) and so on. In popular queries, different people clicked on the same results, so with the increasing query frequency, click entropy partly increases and then remains relatively constant or even in some cases decreases. So the click entropy for popular queries places in a certain range. On the other hand, when a query has high frequency and low click entropy, then the probability that the query is popular is high. In the following, we examine the correlation between click entropy and query frequency along with query features.

In the following, we examine the query features to estimate personalization based on the extracted features from reference [7]. These features include the query length, history, and the clicked results for each query. We begin by examining the simplest features such as query length[11], the query consists of the time, date, place, and query contain the URL. These features are calculated for all queries collected from users using correlation coefficient and are represented in Table 4.1.

We divide other features into the features related to query history and user history. The features that place in the query history category include query frequency (the number of times

[11]The query length is detailed in Figure 0.2., Appendix A.

Query Features	Query Frequency		Click Entropy	
	Low	High	Low	High
Query Length	-0.088	-0.075	0.251	0.395
Query consists of the time-date-place	0.042	0.018	0.150	0.175
Query contains the URL fragment	0.016	0.130	0.192	0.263

Table 4.1. The correlation ratio among features related to the query structure.

query issued), the average of item rank clicked per query, click entropy, and the number of queries with no item rank clicked. All calculations use the correlation coefficients between query features and the measures of query frequency and click entropy. These features are indicated in Table 4.2. The user history features are also calculated in Table 4.2. such as query frequency per user (the number of users who issued query), click entropy query for each user, the number of distinct users, the average of item rank clicked for each query per user.

Query and User History Features	Query Frequency		Click Entropy	
	Low	High	Low	High
Query frequency	1.0	1.0	0.735	0.583
The number of item rank clicked for each query (all combinations)	0.615	0.896	0.060	0.488
The number of item rank clicked for each query(distinct)	0.317	0.038	0.608	0.325
Click entropy	0.735	0.583	1.0	1.0
Query frequency for each user	0.186	0.058	0.022	0.045
The number of users (distinct)	0.726	0.912	0.020	0.464
Average item rank clicked for each query per user	0.021	-0.010	0.274	0.107

Table 4.2. The correlation ratio among features related to the query and user history.

After calculating the properties of the table, we want to discuss the relationships among these measures. The best results and measures are achieved for features that involve query history. To achieve a better result, we have divided these measures to low query frequency and high query frequency.

Based on the calculations in Table 4.2., the features related to query history like query frequency, click entropy and the number of item rank clicked for each query have the highest correlations. As the table shows, click entropy for both low and high query frequency is increasing but in high frequency is less. There is also a dramatic increase in feature the number of item rank clicked for each query. We calculated this feature for both distinct and combinations. There is also a remarkable increase in click entropy for low-frequency queries while a low increase in all combinations. Therefore, it can be inferred that for queries with high frequency, there is less growth to increase for correlations click entropy and the number of item rank clicked for each query while a more growth for low query frequency.

4.2. Estimating Potential for Personalization

In this chapter, the potential for personalization for a global query is explored regardless of the user who clicked on it. We use it before the re-ranking process for each user. The goal is to prevent useless personalization for unambiguous queries regardless of the user. Although there are some metrics such as click entropy and topic entropy to identify queries whether they require personalization, they have limitations especially for queries without history. To overcome these limitations, we present a new metric called unified topic user entropy (UTUE) that estimates the potential for personalization using topic distributions of individual query words. First, we summarize click entropy and topic entropy metrics since we compare their performances with the performance of our new metric.

As discussed in the previous chapter, based on references [6, 7] click entropy is a known metric for estimating personalization in queries. Click entropy measures the query's personalization potential using the clicked documents. If the click entropy is high, it means that different users click on different documents and the query is ambiguous. Click entropy is defined in Equation 1 as the entropy of the documents' click probability distribution for the query. For an unambiguous query, relevant documents are clicked with a higher frequency by different users creating a probability distribution with less uncertainty.

As can be seen from Equation 1, click entropy is purely based on documents but not their contents. When different documents with similar contents are clicked by users for a query q, click entropy will be high signaling a false ambiguous query. Furthermore, in the reference [3] topic entropy discussed in Equation 5 is proposed as a natural extension of click entropy with more performance. In the other words, if the click entropy for a query is high, it means that different users click on different documents and it might be an ambiguous query and as a result potential for personalization in a query will be high.

Topic entropy is the weighted sum of Kullback-Leibler divergences of query and document topic distributions and Yano et al. [3] model the topic entropy as the center of gravity for the topic distribution divergences. While this measure incorporates document similarities, the users' behavioral differences are only modeled through the $P(d|q)$ component. Topic entropy is still not defined (its value is zero) for the new queries the same as the click entropy. Topic entropy on the other hand models $P(d|q)$ using the topic model distribution of the documents, able to account for documents with similar contents. In topic entropy Equation 5 the topic set Z is obtained using LDA. $P(z|q)$ is the probability of the topic z for the given query q and it is estimated using the documents clicked for a query q as in Equation 6.

We try to address these problems in our new metric. Although Yano et al. [3] also propose topic user entropy (TUE) as in Equation 8 to incorporate the users' behavioral differences, in their experiments the correlation of topic user entropy results with human judgments is low compared to Topic Entropy.

$$TUE(q, U_q, D_q) = \sum_{u \in U_q} \sum_{d \in D_q} P(d|q,u) KL(P(z|d)||P(z|q)) \tag{7}$$

$$= \sum_{u \in U_q} \frac{1}{|U_q|} \sum_{d \in D_q} P(d|q,u) \sum_{z \in Z} P(z|d) log(\frac{P(z|d)}{P(z|q)}) \tag{8}$$

Where D_q is the documents clicked for the query q, U_q is the user set issuing the query q. It is assumed that the probability of each user issuing the query is equally likely.

Note that TUE weights the divergence of document model from query model by $P(d|u,q)$ which is the number of times the user u clicks document d for query q, divided by the total number of clicks of u for query q. For a user who did not issue q previously, TUE is not defined since no document is clicked. To solve this cold start problem, we try to benefit from extracted topics of topical user model $P(u|q)$ in our new metric. We define $P(u|q)$ in

Equation 10 and it is the probability distribution of the query on the users using the LDA topic model.

$$P(u|q) \propto P(u)P(q|u) = P(u)\prod_{w\in q} P(w|u) \tag{9}$$

$$= P(u)\prod_{w\in q}\sum_{z\in Z} P(w|z)P(z|u) \tag{10}$$

Where $P(u)$ is the probability of the user u and it is estimated by the proportion of queries submitted by user u to the total number of queries. $P(w|z)$ is the probability of the word w of the query for the topic z and $P(z|u)$ is the probability of the topic z for the given user u. $P(z|u)$ is also estimated using all documents D_u clicked by user u as in Equation 11 and it is used to weight the contribution of each topic for the query.

$$P(z|u) = \sum_{d\in D_u} P(z|d)P(d|u) \tag{11}$$

Using $P(u|q)$ as the weighting factor instead of $P(d|q,u)$, we define our new metric called as the unified topic user entropy (UTUE) as in Equation 13. This metric unifies all users who have or have not issued the query in the past.

$$UTUE(q, U_q, D_u) = \frac{1}{|U_q|}\sum_{u\in U_q} P(u|q)\sum_{d\in D_u} KL(P(z|d)||P(z|q)) \tag{12}$$

$$= \frac{1}{|U_q|}\sum_{u\in U_q} P(u)\sum_{d\in D_u}\prod_{w\in q}\sum_{z\in Z} P(w|z)P(z|u)P(z|d)log(\frac{P(z|d)}{P(z|w)}) \tag{13}$$

As a new query will only be submitted by a single user and will not have any clicked documents, $d_{u,q}$ will be an empty set. As a result, $TUE(q, U_q, D_q)$ will be equal to zero. Instead of depending on the clicked documents for the specific query q, the documents clicked by the user D_u for all queries are used. Furthermore, instead of using $P(z|q)$ which depends on the clicked documents for the query q, the topic distribution of words is used.

4.3. Quantify Query Personalization

Since in the AOL data set, there is not any value to measure personalization, we consider the click and topic entropy as indicators of personalization and calculate the correlation between these metrics[12].

4.3.1. Evaluation Metrics

The potential for the personalization metric is evaluated using a similar methodology to Yano et al. [3]. The correlation between human judgments for query ambiguity and the automatic measures are reported. First for three different frequency levels queries are randomly sampled and two-hundred queries are selected for each frequency level. A total of six-hundred queries is annotated by human judges as “clear”, “broad” and “ambiguous”. Five human annotators are used for the annotations and the inter-rater agreement is estimated using Fleiss Kappa is 0.436.

For evaluation, labeled queries are assigned weights, for ambiguous, broad and clear weights are defined as 2, 1, and 0 values. Then for each query label, scores are calculated as the sum of human-assigned labels. The rank correlation between the human scores and the potential for personalization metrics are calculated using Kendall’s τ.

4.3.2. Experimental Result

We quantify query ambiguity by using discussed evaluation metrics. The performance of the three potential for personalization metrics is investigated using the 600 query ambiguity dataset built in this research. Table 4.3. indicates the correlation using Kendall’s τ in the

[12] The values are detailed in Figure 0.1., Appendix A.

human judgments and three metrics, namely Click Entropy [2], Topic Entropy [3] and the proposed $UTUE$ metric. The results are grouped into three highlighting the effectiveness of the metrics for each frequency category. There are two-hundred queries at each frequency level. The low-frequency group is also separated into three sub-groups to reveal the performance gains of the proposed $UTUE$ metric for low frequencies. If a query's frequency is equal to one, it means that the query is new and its click entropy and topic entropy is equal to zero. 34% of the queries are new without any history where only UTUE can be used for calculating the potential for personalization.

	$\tau forLowFreq$			$\tau forMidFreq$	$\tau forHighFreq$
	$F = 1$	$2 \leq F < 10$	$10 \leq F < 50$	$50 \leq F < 150$	$150 \leq F < 400$
Click Entropy	0.149	0.181	0.252	0.226	0.214
Topic Entropy	0.149	0.268	0.340	0.301	0.283
UTUE	0.297	0.273	0.182	0.207	0.185

Table 4.3. Kendall's τ between methods and ambiguity levels in different query frequency.

The results in Table 4.3. show that $UTUE$ outperforms the topic entropy and the click entropy for the queries without history and the first column indicates that the improvement is doubled for those queries. UTUE's performance for the queries whose frequency is less than 10 is given in the second column where UTUE outperforms the other metrics. However, for queries with frequencies higher than 10, topic entropy and click entropy can estimate the potential more accurately and outperforms the proposed UTUE.

Figure 4.3. shows the plot of frequency and rank correlation for both Topic Entropy and $UTUE$. While topic entropy is more accurate for identifying query ambiguity in higher frequencies, the $UTUE$ is more effective for lower frequencies. It means that the topic entropy measure can successfully determine the ambiguity of a query if there are enough previously clicked for that query. But $UTUE$ performs better for low-frequency queries as it tries to determine the ambiguity of a query from the clicked documents using the individual words forming the query. This result confirms our intuition that $UTUE$ can be used for queries where the other metrics fall short, in queries without a history and for low-frequency queries, and topic entropy should be preferred for the other queries.

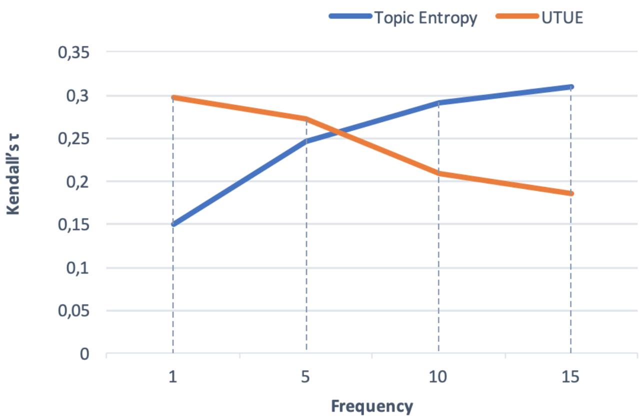

Figure 4.3. The changes in kendall's τ in topic entropy and UTUE metrics for low-frequency queries.

5. USER PROFILING PROCESS

In this thesis, the methodology is divided into three parts: User profiling process, Estimating the potential for personalization, and Re-ranking process for search personalization. A general overview of the framework is described in Figure 5.1. The framework is consists of the main steps of creating the user and group profile, estimating the potential for personalization, and re-ranking results. In the first step, user profiles are created using the keyword/keyphrase based and topic-based structures described below. In the second step, the potential for personalization is estimated using topic entropy metric and proposed new metric(Unified Topic User Entropy) described in Chapter 4. Finally, in the last step, the personalization process completes with ranking the result using user and group profiles and combinations methods described in Chapter 6.

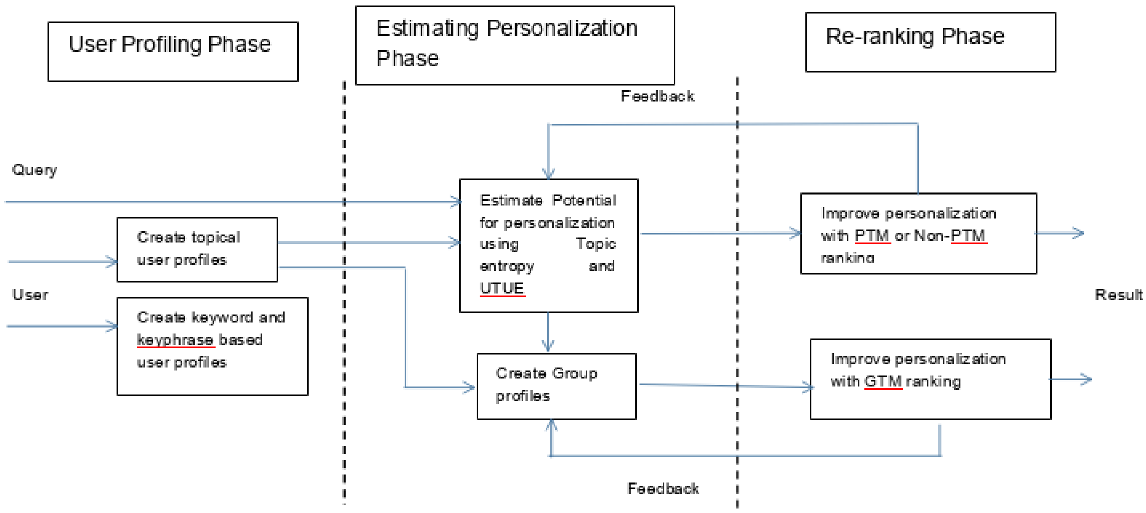

Figure 5.1. An overview of the framework includes the main steps.

As mentioned before the purpose of the user profiling process is to collect user information based on user needs and interests. There is an overview of the user profiling process in personalization [21, 248]. The system developed in [18] has divided the user profiling process into three main works that consist of data extraction, data integration, and discovery. In a similar approach in [112]the authors added one additional phase visualization to the process.

Zhiheng et al. [249] proposed a multi-interest modeling method to complete a hierarchy structure to improve efficiency. Based on their research the scheme is consisting of three modules: user modeling module for managing data, text classification training module, and semantic similar network training module.

A user profiling process can be conducted into the following subprocess, collecting information such as experience, interests, behaviors, pre-processing of collected information, and creating a user model. Finally, the process is completed using applications to provide personalization services. There are a number of most popular techniques in collecting, representing, and creating user models [74]. In this chapter, we describe the techniques used.

5.1. Collecting Information

User profiling is started with collecting information about the users. Older systems gathered data explicitly from the users by asking the users, but the user is not interested in directly giving information. Besides, this method is needed to fill a lot of questions and may not always provide correct answers.

Nowadays more research is focused on implicit methods or behavioral user profiling for collection information. There is two major matter of collecting information. The content of the data collected and techniques to collect information. Since in this thesis, we use implicit methods to collect user information, so we discuss in detail in the following.

5.1.1. Techniques to Collect Data

There are several techniques to capture the user's interests using exploring his/her actions and inferring interests and preferences such as browser caches and agents, history and browsing search, proxy servers, and search logs. Some of the techniques collect the user activities while interacting with the site on the side of the server, for example in browser cache and proxy server methods. There are two methods for collecting browsing histories, one is installing a proxy server by users and the other is sharing the browsing caches periodically by

Example publications	Server side	Client side	Implicit	Explicit
Asnicar and Tasso [185]		X		X
Chen and Sycara [153]		X		X
Micarelli and Sciarrone [189]		X		X
Speretta and Gauch [186]	X			X
Chirita et al. [149]		X		X
Teevan et al. [1]		X	X	
Chirita et al. [212]		X	X	
Tao et al. [250]	X		X	X
Bennett et al.[150]	X			X
Huang et al. [99]		X	X	
Hannak et al. [251]	X		X	X
White et al. [152]	X			X
Yang et al. [159]	X			X

Table 5.1. The techniques used to collect data in some example publications.

users. In browser agents, the agents can collect data with browsing history by bookmarking, downloading, scrolling, and moussing.

Web and search logs collect the browsing and query histories to build user profiles to help personalization. This method is not required to install a desktop application by users on the client-side such as Google Desktop Search. Table 5.1. shows the techniques used to collect data in some example publications. In this thesis, we use a query log collected by the AOL search engine and session TREC 2014.

5.1.2. The Content of the Information Collected

There is a range of content information such as click-through data or query log analysis, mouse movement, scrolling, browsing the history, desktop information analysis, dwelling or display time, etc. For example, there are some studies with a focus on created user profiles by the content of browsing history [43, 61, 153, 185]. As discussed in Table 5.1. Chen and Sycara [153] have used multiple TFIDF vectors for creating the WebMate system. Pretschner et al. [61] also used browsing history and user preferences are created over time. Sugiyama et al. [43] and Gauch et al. [185] have built a user profile based on browsing history to re-rank results from a search engine. Also, Lieberman [162] with the Letizia system, Mladenic

[165] with the Personal Web Watcher system and Gauch et al. [73] with OBIWAN, have built the user profile by analyzing the browsing history.

The authors [39, 46, 252] have used log files and click history to learn user profiles. Hijikata et al. [253] have traced the user's mouse operation and Thomas [254] has tracked the user bookmarks to create the profiles. Collecting data from different sources such as web pages, email, and documents are used by Dumais [255]. Matthijs et al. [256] collected the data from several components such as the visited pages in the search process and the time opening on the page and Xueping [99] used the user search result concerning Google Directory.

The Honeypots is introduced by Spitzner [257] to capture user information. The Honeypots is designed as a website that attracts the users and is tracked internet usage patterns of the users. Social networks can be also used to collect information about users for example Jiwei [258] has extracted user information from Twitter, Google Plus, and Facebook. In this research, the author used a supervised approach to extract user profiles from Twitter. Kelly et al. [38] have used implicit feedback to recommend a suitable web page to the user. The paper is used some special implicit feedback such as text tracing and selection, link pointing, printing a page, window movement, and resizing. Generally, collection information using social media and networks is more accurate. In other words, many social media help to build user profiles [259–266]. For example, Guy et al. [259, 260] have built a personalized recommendation using social media, and Chen [262] used Twitter for the personalized recommendation. The conversations are used to build user profiles of friends [263] and communities [265].

The online discussion or forums like Digg, are a type of social media where people gather together and discuss a specific topic for getting or sharing information with others. In forums, user's interests are showed by provided content and the exchanged opinions in discussions. For example, the authors [98] created a hierarchical user profiling based on generated content and the topics of the discussions by the users in forums.

Extending on this idea, Ghulam [60] presented a model that uses of social agents and user activities on the internet to create a community of similar interests. Nazim et al. [111] also used the social web to collect information to create the profiles. So, the first commitment of a system is the method of collecting data for the user model and the second is to uniquely identify users.

According to Susan Dumais et al. [6] research, clicked item rank and URL by users are the most critical feedback used by researchers for the personalization field. So, in this research, we used the click data collected by users. We used a query log collected by the AOL search engine as a large amount of click-through. Using implicit data sets, we can study many different users' interactions while it will be infeasible with explicit data.

5.2. Pre-Processing of Collected Information

After the collection of relevant information, it might preprocess to remove duplicate and clean up the data. Before cleaning the data set, we should uniform the language in available data. Since the query log is obtained from a search engine with different geographies, we only retain the queries in English. Then, we cleaned the dataset by only retaining some queries. It means that are dropped queries without clicked results. Then the data is filtered by removing URLs clicked less than one-hundred times.

Besides, the extracted URL or document also needs to be cleaned. To clean them, we extracted clicked pages by users and re-open them using a search engine. Then we did a pre-processing on extracted documents. To do, we did a collection of steaming, stop word removing, and parsing process. For steaming, we applied porter steaming on words on pages.

5.3. Start to Create User Models

In this step, we want to begin to create user models with the collected and preprocessed information from users. As we mentioned before, there are some approaches and techniques to represent and structuring user profiles. In this part, by investigating user profile structure, we create user profiles as keyword-based, keyphrase based and topic-based, and in the following, we explain both them in detail.

5.3.1. Keyword and Keyphrase based User Profiles

We create profiles as keywords and keyphrases extracted of clicked documents by users using keyword/keyphrase extraction methods. We use a number of models such as TFIDF, RAKE[13][267], TopicRank [268], TextRank [269], KEA(Keyphrase Extraction Algorithm) [270] and WINGNUS [271] to extract keywords/keyphrases. In the process of keywords/keyphrases extraction, there are supervised and unsupervised methods. Current methods in unsupervised are divided into statistical models and graph-based models. Among statistical models, TFIDF is used successfully and we intended to use it as a basic algorithm to the comparison. TFIDF is described in Equation 14 is a weighting technique used in information retrieval. In search engines, TFIDF is used to measure the similarity between a query and a document. The Tf computes the frequency of a word. The idf estimates the logarithm of the number of the documents divided by the number of documents with the specific word.

$$w_{i,j} = tf_{i,j} \times log\frac{N}{df_i} \tag{14}$$

Where $tf_{i,j}$ is the number of occurrences of i in j , df_i is the number of documents containing i and N is the total number of documents. Since the length of documents is different, so the term frequency is often divided by the document length as a way of normalization. In idf also some terms with low importance have appeared a lot of times. So, we need to weigh low the frequent terms.

In keywords/keyphrases extraction algorithms, in the first step, keywords candidates are extracted and then the extracted keywords are sorted by a weighting method or a machine learning technique as unsupervised or supervised. Finally, the top-K highest weighted candidates are selected. More on unsupervised keywords/keyphrases extraction methods, we apply efficient graph-based models such as RAKE, TopicRank, and TextRank. These algorithms are based on the graph-based modeling which a graph is built based on words or phrases where the edges' weights are computed using co-occurrence counts [272, 273].

[13]Rapid Automatic Keyword Extraction

The RAKE algorithm proposed in [267] is a language-independent method that extracts keywords by analyzing word frequency and its co-occurrence with other words in the text. RAKE focuses on finding multi-word phrases containing frequent words. First, RAKE splits the text into sentences using punctuation signs and generates the candidates. All words listed in the stop-word file will be treated as phrase boundaries. This helps generate candidates that consist of one or more non-stop words, such as 'compatibility,' 'systems,' 'linear constraints,' 'set,' 'natural numbers,' and 'criteria' in this text. RAKE is based on the theory that key phrases frequently contain multiple words but rarely contain standard punctuation or stop words or other words with minimal lexical meaning. Most of the candidates will be valid phrases; however, it won't work in cases where the stop-word is part of the phrase.

RAKE focuses on finding the most frequent phrases with multiple words. Removing punctuations and stop-word from the text is the primitive step. Then the properties of each candidate are calculated and sorted according to frequency or RAKE's scores. The advantages of RAKE are simplicity and ease of use while there are some limitations such as accuracy, the lack of stemming, and normalizing keywords the parameter configuration requirement. Besides the Rake doesn't yield the correct result for words include stop-word.

TextRank algorithm proposed in [269] is an algorithm based on PageRank and the same as RAKE extracts key phrases by a co-occurrence graph. In this algorithm, documents are split into the sentences, and the words with specific tags such as noun, prop, and verb. Then a graph of words is created and the weight for each node is calculated. Then it collects the influence of each of its connections and determines the new score for the node. In this way, TextRank considers the similarity between each sentence to all other sentences.

TopicRank algorithm proposed in [268] is an improvement of the TextRank that extracts keyphrases using a topical representation. Extracted keyphrases are clustered into topics as a graph-based structure.

Given a complete graph $G = (V, E)$ where V and E are a set of vertices and edges. The topics are shown as nodes and the weight $w_{i,j}$ between two nodes is measured as Equation 15 defined [268].

$$w_{i,j} = \sum_{c_i \in t_i} \sum_{c_j \in t_j} dist(c_i, c_j) \tag{15}$$

$$dist(c_i, c_j) = \sum_{p_i \in pos(c_i)} \sum_{p_j \in pos(c_j)} \frac{1}{|p_i - p_j|} \tag{16}$$

Where the distances between the candidate keyphrases c_i and c_j is measured by Equation 16 and $pos(c_i)$ represents all the offset positions of the candidate keyphrase c_i. After creating a graph, the ranking model TextRank, is used over the topics. The ranking model uses a score function as Equation 17 defined by reference [268] to sort the topics.

$$S(t_i) = (1 - \lambda) + \lambda \times \sum_{t_j \in V_i} \frac{w_{i,j} \times S(t_j)}{\sum_{t_k \in V_j} w_{j,k}} \tag{17}$$

Where V_i and λ are the topics voting for t_i and a damping factor generally set to 0.85.

In Chapter 7. to complete a benchmark, we will evaluate supervised models like KEA and WINGNUS as baselines methods. KEA builds a classifier based on the Bayes'theorem using training documents, and it uses the classifier to extract keyphrases from new documents. In the training and extraction phases, KEA analyzes the input document depending on orthographic boundaries (such as punctuation marks, newlines, etc.) and exploits two features: tfidf and the first occurrence of the term. The WINGNUS similar to KEA uses a Naive Bayes classifier and POS rules to extract keyphrases. The extracted keyphrase are stored in the keyphrase based user profiles in the form of XML and will be applied in the ranking process.

5.3.2. Topic-based User Profiles

Although keywords/keyphrases extract the key topics, they cannot analyze the content of the document. The most simple way to build a topic-based profile is that use an online ontology produced by humans like mentioned before. However, this approach has limitations such as the lack of online categorization related to a document and pre-determining the categories for each document by a human. The approaches such as LDA, LSA, PLSA, and LDA2VEC try to extract hidden topics in a text.

To create a topic-based user profile, it needs to extract topics from relevant documents. As a result, the topics are hidden or latent concepts that are explaining its document. In this thesis, in particular, we use the LDA an unsupervised learning method, and a Bayesian version of pLSA. The LDA uses the original Dirichlet distributions to distribute the topics over the document and words.

The LDA using a set of documents m, in the first step, assumes the hidden k topics and distributes the topics across these documents. In the next step, for each word w in the document assigns the correct topic. Finally, for each topic specifies the words based on the documents and the frequency of words. The LDA model is depicted in Figure 5.2..

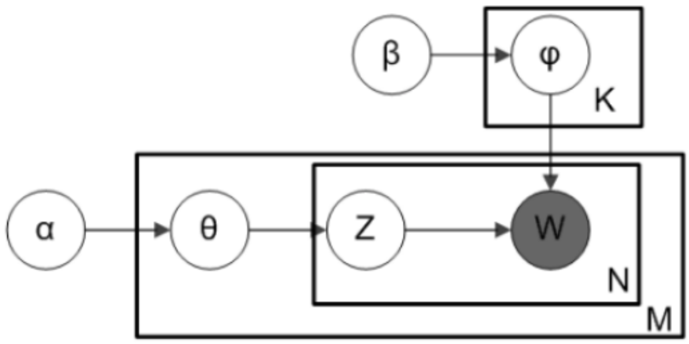

Figure 5.2. The LDA algorithm.

Figure 5.2. depicted an LDA model with α, β and φ as hyperparameters according to document, word and topic distribution for document m, topic z and word w. The LDA is the most popular and effective topic modeling technique that can be easily used on new documents. It is available in the Gensim library as Figure 5.3. The LDA is used in the fields such as text segmentation [274], tag recommendation [275], automated essay grading [276], topic identification [277], and web spam classification [278].

```
from gensim.corpora.Dictionary import load_from_text, doc2bow
from gensim.corpora import MmCorpus
from gensim.models.ldamodel import LdaModel

document = "This is some document..."

# load id->word mapping (the dictionary)
id2word = load_from_text('wiki_en_wordids.txt')

# load corpus iterator
mm = MmCorpus('wiki_en_tfidf.mm')

# extract 100 LDA topics, updating once every 10,000
lda = LdaModel(corpus=mm, id2word=id2word, num_topics=100, update_every=1, chu

# use LDA model: transform new doc to bag-of-words, then apply lda
doc_bow = doc2bow(document.split())
doc_lda = lda[doc_bow]

# doc_lda is vector of length num_topics representing weighted presence of eac
```

Figure 5.3. Implementation of LDA using Gensim.

In this thesis, we need to create topic models of extracted documents of clicked URLs by the user. To create topic models, we use LDA that involves an iterative Bayesian topic assignment process over a train-corpus. The topic extracted can be also obtained by using an online ontology that has some problems. Thus we create topic models by deriving topics from the data itself as the work done by Harvey et al. [4].

A personalization topic model is depicted in Figure 5.4. To build the model, the number of topics needs to be preset. The prior distribution of topics over documents is taken as Dirichlet. The model involves a document d, a topic variable z, a word w, and a user u.

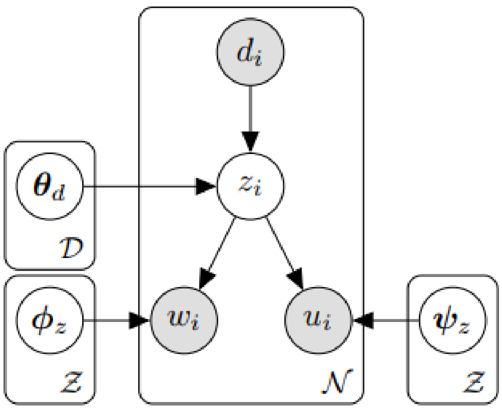

Figure 5.4. A personalization topic model.

As we mentioned before, we can get to the topic distribution across documents and the words included in the topics assuming the dependency between the user and the words to the topic. In this model θ_d, ϕ_z, and ψ_z are defined as probability vector over topics, words, and users in the modeling process. In addition, α, β, and γ are defined to uniform distributions for sparse data and removing the overfitting problem. The posterior distributions $\hat{\phi}_{w|z}$, $\hat{\theta}_{z|d}$ and $\hat{\psi}_{u|z}$ are described using Equations 18, 19 and 20 that are taken from the research conducted by Harvey et al. [4, 5].

$$\hat{\phi}_{w|z} = \frac{N_{w,z} + \beta \frac{1}{W}}{N_z + \beta} \tag{18}$$

$$\hat{\theta}_{z|d} = \frac{N_{z,d} + \alpha \frac{1}{Z}}{N_d + \alpha} \tag{19}$$

$$\hat{\psi}_{u|z} = \frac{N_{u,z} + \gamma \frac{1}{U}}{N_z + \gamma} \tag{20}$$

Where $N_{w,z}$, $N_{z,d}$ and $N_{u,z}$ are the number of times the topic z appears together with the word w, document d, and user u respectively. The common methods recommended to approximate the posterior distribution are variational inference [117] and Gibbs sampling [277]. We used Gibbs sampling based on the Markov chain approach in our work.

In our model, each state of the Markov model indicates the allocation of topics to words of search queries, and this process continues to reach a satisfactory convergence. After reaching an acceptable convergence, the parameters can be estimated.

As such, we begin by creating the user topical profiles that the documents are modeled using the distribution over topics as Figure 5.5. The parameters will be explained in the evaluation metrics in Chapter 7. Using the distributions explained in Equations 18, 19 and 20 we will construct a ranking model to sort the documents. The details related to re-ranking implementation are discussed in Chapter 6.

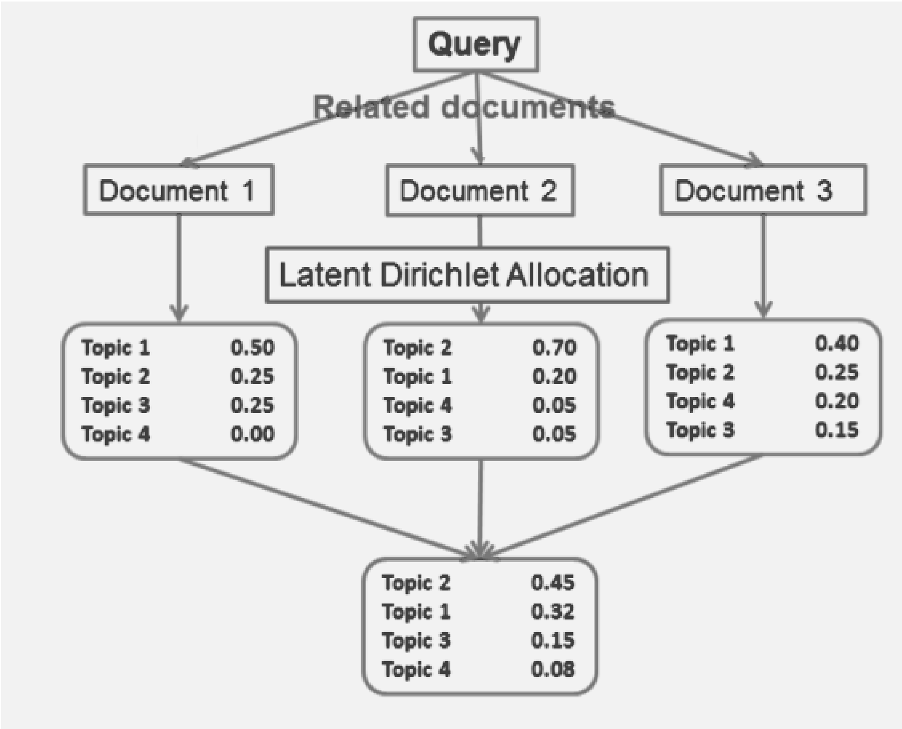

Figure 5.5. LDA model using related clicked documents

5.3.3. Group User Profiles

The user profiles can be enriched by adding the information of other users with common interests to the user profiles, and it can improve the efficiency of web search results [2, 47, 152]. These group profiles are created based on the common interest of groups of users. These group profiles can be created on the keyword/keyphrase-based profiles or topic-based profiles. In this research, we create group topical user profiles to the purpose of comparing topical user profiles and we present the experiments in the evaluation chapter.

5.3.4. Building Persistence and Temporal User Profiles

As time is changing, user profiles are also changing. In some studies [43, 114] are used the different vectors to model the user interests as short-term and long-term. In another study by Bennett et al. [150] the weights of clicked URLs are used to represent how interests change over time. We also consider time in creating user profiles. In this way to build short-term and long-term user profiles, we consider topical user profile for one and three months. In this order, we can consider the topical user profile as sessions of time. Table 5.2. shows some publications of several approaches in the literature as short-term and long-term profiles.

Example publications	Long-term	Short-term
Nanas et al.[148]	X	
Mc Gowan [56]	X	
Dou et al. [2]	X	
Shen et al. [141]		X
Sugiyama et al. [43]	X	X
Chirita et al.[149]	X	
Teevan et al. [1]	X	
Sontag et al. [151]	X	
Harvey et al. [4]	X	
White et al. [152]		X
Bennett et al. [150]	X	X

Table 5.2. Some publications of several approaches in the literature as short-term and long-term profiles.

To build user profiles, we used the AOL and TREC 2014 Session Track data set. We separated documents for each user in the query log and then create user profiles in two representing methods of keyword/keyphrase-based and topic-based. Despite our focus on topic-based user profiles, we create keyword/keyphrase-based profiles to compare the ranking performance result.

6. RE-RANKING PROCESS FOR SEARCH PERSONALIZATION

Personalization is the task of re-ranking the retrieved document set concerning the user profile. While this task on its own is independent of the potential for personalization tasks, we argue that selective re-ranking can yield better results.

The first we discuss two ranking methods include ranking based on the keyword/keyphrase profiles and ranking based on the topical user profiles. The purpose is to make a comparison between keyword/keyphrase based models and topic based models. Since our focus is on the topical model and topical user profiles, three ranking approaches based on the topical model are used in our evaluations, where the first one uses a generic document scoring function based on topic models without any personalization. The second model uses the personalization factor using the documents clicked. The last approach re-ranks the result using a group based topical profile.

Although profiles are indicative of the user interests, they can be incomplete and misleading due to data sparsity. For user profiles with less browsing history, data available might not be sufficient. To resolve the data sparsity, the history of similar users can be grouped and used for personalization. As a final re-ranking model, we propose a new group based personalization method.

6.1. Ranking based on Keyword/Keyphrase based User Profile

In this method, the documents are ranked using keyword and keyphrase based user profile. In the re-ranking process, unlike some papers that benefit from Google API results, we use the clicked documents by the user. In other words, we use the query log to rank documents. Re-ranking documents for a new query using keyword and keyphrase based user profile, we measure the similarity between the extracted keywords/keyphrases of each document in the query log and the keyword/keyphrase based user profile for each user by cosine similarity measurement. This measurement is used to measure changing between similar documents regardless of the size of the document. It can be measured using Equation 33. For the evaluation re-ranking process using keyword and keyphrase based user profile, we tried a re-ranking algorithm on the AOL and TREC 2014 Session Track data set and the results are presented in Chapter 6.3.1.

6.2. Ranking based on Topical User Profile

6.2.1. Generic Ranking without Personalization

Given the LDA topic models, documents, and words are associated with topics in the document set. Building on the same framework introduced by Harvey at al. [4], documents are ranked using the LDA model $P(d|q)$ called NonPTM (Non-Personalized Topic Model) here. The $P(d|q)$ is estimated using the Bayes rule and the LDA generative model as follows.

$$NonPTM(d,q) = P(d|q) \propto P(d)P(q|d) = P(d)\prod_{w\in q} P(w|d) = P(d)\prod_{w\in q}\sum_{z} P(w|z)P(z|d) \tag{21}$$

Based on the described LDA model, we build a topical user profile for each user in the data set using the distributions obtained. Based on the estimated likelihood in paper [3], we rank the documents in the data set for unpersonalized (LDA) model as Equation 21.

Where $P(d)$ is the prior document probability and z is the topic latent variable estimated using LDA. $P(w|z)$ and $P(z|d)$ are obtained from the LDA topic model. Since NonPTM is a method without any personalization, comparisons with this baseline method will reveal the improvement of personalization over generic ranking with topic models.

6.2.2. User Profile based Personalization

The personalization method of Vu et al. [171, 279] is reproduced for completeness. A user topical profile is modeled by a set of documents D_u which the user clicked on. Using the topic distributions of the user's documents that are associated with topics, the user profile can be considered as the vector of posterior probabilities of topics given the user and is calculated as in Equation 22.

$$P(u|z) \propto P(u)P(z|u) = P(u) \sum_{d_i \in D_u} P(z|d_i) \tag{22}$$

Then, the personalization based ranking function is defined as in Equation 23 which will be referred to as Personalized Topic Model (PTM).

$$PTM(d,q,u) = P(d|q,u) \propto P(d) \prod_{w \in q} P(w,u|d) = P(d) \prod_{w \in q} \sum_{z} P(w|z)P(u|z)^{\lambda} p(z|d) \tag{23}$$

The λ parameter weighs the effect of user topical profile on the ranking process and it is equal to 0.175 similar to Harvey at al. [4]. That $\lambda \in [0,1]$ is used to weight the probability of $P(u|z)$ and to calculate the probability of a user given a particular topic or topical user profile we use the Equation 24.

$$P(z|u) = \frac{1}{|D_u|} \sum_{d \in D_u} P(z|d_i) \tag{24}$$

6.2.3. Temporal User Profiles

Since during a search session, user interests and search intentions are changing, so the long-term and short-term profiles were also discussed [150, 280]. For example, Vu et al. [280]

created temporal user-profiles and used for the re-ranking process and in the similar research Bennett [150] built different temporal user-profiles and the re-ranking results were improved using metrics like click entropy.

There is much research [158, 280] in creating user model into ranking algorithms in various areas effectively but more research has a problem that they have ignored that the user interests change over time. In fact, as time goes on, the user becomes reluctant to some topics while starting attention to other topics.

Considering time sessions in user profiles, similar to Vu et al. [279], an exponentially decaying function is used in order to set more weight to recently clicked documents and t_{d_i} is equal to 1 for the most recent relevant document for the exponential decay function penalizing older clicks. The accumulated evidence is transformed into a probability using the K normalization function calculated as the sum of document biases $K = \sum_{d_i} \alpha^{t_{d_i}-1}$ and the α parameter of the decaying function is set to 0.95 the same as Vu et al. [279]. Then, the personalization based ranking function is defined as in Equation 23 where $P(u|z)$ is estimated as Equation 25.

$$P(u|z) \propto P(u)P(z|u) = P(u)\frac{1}{K}\sum_{d_i \in D_u} \alpha^{t_{d_i}-1} P(z|d_i) \tag{25}$$

6.2.4. Group Profile Personalization

The function $PTM(d, q, u)$ which is reproduced from Vu et al. [279] depends on users and their topic distributions estimated using the documents clicked by the users. One disadvantage of $PTM(d, q, u)$ is that it is built by considering only the documents that the user u has clicked and the set of the clicked documents might be sparse for some users.

Data sparsity can be resolved by backing off to the group of users with similar behavior to the user u. We propose a group profile based personalization method which first groups users with respect to their topic distributions and use group profiles in the ranking process.

Users are clustered using their $P(z|u)$ topic probability distributions. Topic probability distributions depend on the documents that are clicked by users and they are estimated with Equation 24. K-means clustering algorithm is used to partition the users to $|C|$ groups. The number of clusters $|C|$ is a parameter and C_u is the cluster of user u. Our proposed group profile based personalization method called a Grouped Personalized Topic Model (GPTM) determines the ranking score with respect to the group profiles and it is defined in Equation 27.

$$GPTM(d,q,u) = P(d|q,C_u) \propto P(d)\prod_{w\in q} P(w,C_u|d) \tag{26}$$

$$= P(d)\prod_{w\in q}\sum_{z} P(w|z)P(C_u|z)^{\lambda}P(z|d) \tag{27}$$

Equation 27 generalizes the user ranking to the clusters of users, resolving the sparsity problem. The λ parameter weighs the effect of group profile on the ranking process and group profiles are computed as follows.

$$P(C_u|z) \propto P(C_u)P(z|C_u) = P(C_u)\frac{1}{K}\sum_{d_i\in D_{C_u}} \alpha^{t_{d_i}-1}P(z|d_i) \tag{28}$$

The computation of group profiles is similar to the computation of user profiles except that the documents that are clicked by all users in a cluster.

6.3. Experimental Results

6.3.1. Re-ranking based on Keyword/keyphrase-based User Profiles

Using the ranking methodology discussed in Chapter 6.1., the performance of personalized profiles are evaluated. This chapter is organized to highlight the key findings of the comparison of the efficiency in keyphrase algorithms using the open-source toolkit PKE[14] consist of supervised and unsupervised keyphrase extraction methods.

All keyphrase extraction approaches are used by the pke toolkit except RAKE. For extraction keyphrase using RAKE, we used a modified version of Python implementation for RAKE. To use it with a specific language, it can be set parameters however in this research, we used the English language. It can be also controlled the maximum or minimum words in a phrase, to obtain a better result by modifying the number of extraction keyword in the rake algorithm. To investigate the effect of various parameters in rake, we explored these parameters depend on the text. It has shown in Figure 6.1. the parameters that performed best on the current dataset with at most three keywords in each phrase.

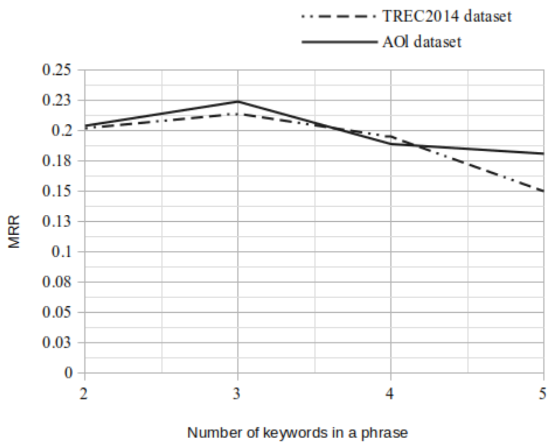

Figure 6.1. The MRR changes in phrase length in words in Rake keyphrase extraction in AOL and Session Track 2014 dataset.

To evaluate the proposed model, we provided two experiments with different datasets. In the first experiment, we used the AOL query log data set as a large resource of explicit query log

[14]https://github.com/boudinfl/pke

	PKM-TextRank			PKM-Rake			PKM-Tfidf		
	S@1	S@10	MRR@10	S@1	S@10	MRR@10	S@1	S@10	MRR@10
Long-Term	0.201	0.353	0.259	0.181	0.310	0.224	0.172	0.308	0.218
Short-Term	0.280	0.410	0.325	0.250	0.368	0.297	0.255	0.358	0.296

Table 6.1. Ranking performance of personalized keyphrase-based methods on the AOL data set.

data. In both the data sets clicked documents are a reference in building user profiles. We will present the result of the experiments in the following.

To build user profiles, we collected documents for each user in the query log and then used keyphrase extraction algorithms described to extract keyphrases. To investigate the importance of keyphrase extraction in personalization, different keyphrase methods are implemented to compare performance in the ranking process. The models are called as personalized keyphrase extraction(PKM). For example, PKM-TextRank is a personalized ranking model based on TextRank keyphrase extraction. In this model, TextRank keyphrase extraction is used in creating user profiles.

Tables 6.1. and 6.2. report the MRR, $S@1$, $S@10$ and $nDCG@10$ scores for the methods for personalization in AOL and Session Track 2014 datasets. In Table 6.1., the columns are represented the used methods including TextRank, Rake, and Tfidf. The Tfidf is used as a basic method to compare the result. To consider the time dimension in the process of user profiling, we separated profiles as short-term and long-term user profiles. As depicted in Table 6.1., the best results are obtained with the TextRank method. In Personalized using the PKM-TextRank model, the MRR is 32% for the short-term and 26% for the long-term while it takes more time to run rather than Rake and Tfidf. It can be seen in all methods, the short-term user profiles are more efficient than the long-term user profiles.

Table 6.2., represents the ranking score when using personalization based on keyphrase based profiles using a set of supervised and unsupervised approaches. Among supervised approaches, KEA and WINGNUS are used as basic approaches while the best results ($\%35$ in $nDCG@10$) are obtained using created models using these methods. In supervised methods also we tried TextRank, Rake, and TopicRank as graph-based methods and tfidf as a

Personalized models(PKM)	S@1	S@10	MRR@10	DCG@10
PKM-TextRank	0.197	0.318	0.238	0.263
PKM-Rake	0.175	0.303	0.214	0.249
PKM-TopicRank	0.204	0.326	0.240	0.272
PKM-Tfidf	0.213	0.335	0.257	0.280
PKM-Kea	0.257	0.390	0.304	0.348
PKM-WINGNUS	0.260	0.395	0.310	0.352

Table 6.2. Ranking performance of personalized keyphrase-based supervised and unsupervised methods on the Session Track 2014.

feature-based unsupervised method. As shown in Table 6.2., PKM-Tfidf resulted in the best performance in $nDCG@10$ among all approaches.

To investigate the effect of the time factor in creating a user profile on the SessionTrack dataset, we compared session-based and long-term based profiles. The result is depicted in Figure 6.2. As the result shows supervised approaches (KEA and WINGNUS) have provided more successful results although those are needed for the additional training step. The tendency in performance in session-based profiles is more than long-term profiles in all personalized models.

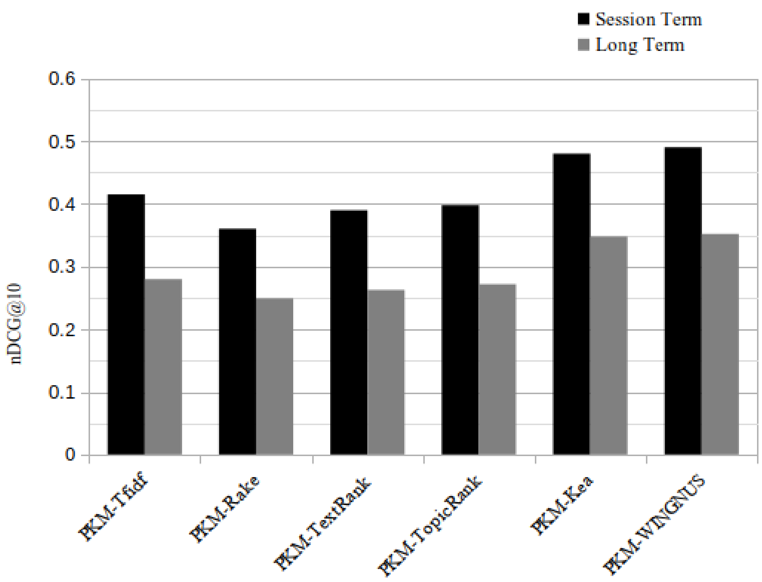

Figure 6.2. Ranking performance of the six models PKM-tfidf, PKM-Rake, PKM-TextRank, PKM-KEA ,PKM-WINGNUS and PKM-TopicRank using $nDCG$ metric on the session and long-term user profiles.

6.3.2. Re-ranking based on Topic-based User Profiles

We re-rank the documents in our data set using created topic models and the results are presented in Table 6.3. as Personalized Topic Model(PTM) and Non-Personalized Topic Model (Non-PTM). Table 6.3. indicates the results for the personalized and unpersonalized ranking methods. Based on the results, ranking based on Topical user profile does not make a large improvement in ranking. While we expect more improvement in the personalized model but there is the same result for non-PTM and PTM. Therefore, there is a need for further examination of the results.

It reveals the need to use personalization metrics to distinguish when personalization should be used for a query and conversely when do not should be used the personalization. In the evaluation chapter, we complete our research and present the result.

	Non-PTM			PTM		
	S@1	S@10	MRR	S@1	S@10	MRR
Long-Term	0.205	0.371	0.267	0.206	0.387	0.272
Short-Term	0.282	0.436	0.330	0.301	0.478	0.366

Table 6.3. Ranking performance of Non-PTM compared to PTM method on the AOL data set.

7. EVALUATION SEARCH PERSONALIZATION

7.1. Preparing the Datasets

In this thesis, we use the click-through data because it is available a large volume of data of different users' interactions in implicit data, while the explicit data only represents a few data about users.

A dataset of web search engine logs of AOL[15] is proposed to investigate the effectiveness of the proposed methods. The Query Log dataset contains three-month data starting from March 2006. The data set consists of an anonymous user ID, Query words, Query Time, the rank of the items on which the user clicked on, and Click URL. In the dataset, there are two types of events consist of a submitted query by a user and user clicked item rank. To clean the data first, we filter the queries with language except in the English due to uniform geographic and linguistic problems. Then as it is done by Harvey et al. [4], we cleaned the dataset by only retaining usable some queries. It means that are removed the queries without resulted in a click. Then the data is filtered by removing URLs clicked less than one-hundred times. The extracted dataset is shown in Table 7.1.

As a second dataset, the TREC 2014 Session Track[16] data is used for the experiments. The Session Track consists of 1021 query sessions for 60 different topics along with the clicked documents and user ids. The URLs are manually annotated by judges for the topics as spam (-2), not relevant (0), relevant (1), highly relevant (2), key (3), and navigational (4). We use the content of the clicked URL to create topic models of user profiles. The extracted dataset is shown in Table 7.1. To evaluate the personalized model, we divided the dataset into 95% for training and the last 5% of queries for testing.

[15] American web portal and online service
[16] https://trec.nist.gov/data/session2014.html

	#Queries	#Users	#URLs
AOL Data set	1,452,012	4,217	11,209
TREC 2014 Session Data set	2550	148	1097

Table 7.1. Extracted data from AOL and TREC 2014 Session data set for experimentation.

7.2. Evaluation Metrics and Methodology

Evaluation metrics in retrieval systems are a critical part of the process of evaluating the results. In the process of evaluation of the results, the focus is on the measurement of the degree of matching between the submitted query by a user and the results obtained by an information retrieval system. A simple approach to evaluate the accuracy of matching the results is to use humans to label the results. This approach has limitations.

To evaluate the personalized models, using the described datasets, we divided the datasets into 95% for training and the last 5% of queries for testing. This chapter is organized to highlight the key findings for these metrics and measuring the performance of the proposed method.

7.2.1. Precision at Rank k

In the process of information retrieval with a ranked list as a returned result, the top-n results are the first n in the ranking. The $precision\ at\ rank\ k (P@k)$ metric measures how accurate are your predictions. The $precision\ at\ rank\ n$ is calculated as Equation 29 using the proportion of the top-n relevant results. In this Equation r is the relevant documents at rank n.

$$P@n = \frac{\sum_{r=1}^{n} rel(r)}{n} \tag{29}$$

Success at rank k is the proportion of recommended items in the top-k (Here $k = 1, 10$) set that is relevant. Where $rel(r)$ is a binary function with two items, one and zero, in which one shows that the document is relevant to the query and zero shows that is irrelevant.

$$rel(r) = \begin{cases} 1, & if\, r \in R \\ 0, & otherwise \end{cases} \tag{30}$$

7.2.2. Mean Reciprocal Rank & Mean Average Precision

The Reciprocal Rank (RR) calculates the reciprocal of the rank at which the first relevant document was retrieved as Equation 31. It means that RR is equal to 1 when a relevant document was retrieved at rank 1 and is equal to 0.5 when a relevant document was retrieved at rank 2 and so on. The Mean Reciprocal Rank (MRR) is also obtained by calculating the average across queries.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{31}$$

Where Q shows the number of queries and $rank_i$ is the rank of document d for query q obtained from the ranking model. When there is only one relevant document in the list of results, the MAP is calculated as the MRR.

7.2.3. Normalised Discounted Cumulative Gain at Rank k

$Normalized\ Discounted\ Cumulative\ Gain(DCG)$ is a measure of ranking quality discussed in [281] and measures the gain, of a document based on its position in the result list. The $nDCG@k$, is calculated over the top k results of search engine as Equation 32:

$$DCG(S,k) = \sum_{j=1}^{k} \frac{2^{r(j)} - 1}{log(1+j)} \tag{32}$$

Where the k is a particular rank threshold, the $r(j)$ is the judgment function at rank j in set S, and S is a set of ranked results.

7.2.4. Cosine Similarity

Cosine similarity is a metric to measure the similarity between the documents based on the number of common words using the Euclidean distance equation of two vectors. It is useful because of measures changing between similar documents regardless of the size of the document. It can be measured using Equation 33 as:

$$cos(\theta) = \frac{\vec{a}.\vec{b}}{\parallel \vec{a} \parallel \parallel \vec{b} \parallel} = \frac{\Sigma a_i b_i}{\sqrt{\Sigma a_i^2}\sqrt{\Sigma b_i^2}} \tag{33}$$

Where $\vec{a}.\vec{b} = \Sigma a_i.b_i = a_1 b_1 + a_2 b_2 + ... + a_n b_n$.

7.2.5. Kullback-Leibler Divergence

Kullback-Leibler Divergence measures the similarity between two probability distributions P and Q while by calculating the cross-entropy minus the entropy as Equation 34 that $H(P,Q)$ is cross-entropy and $H(P)$ is entropy and are estimated as Equation 35 and 36.

$$D_{KL}(P \parallel Q) = H(P,Q) - H(P) \tag{34}$$

$$H(P,Q) = \text{Æ}_{x \sim P}[-log(Q(x)] \tag{35}$$

$$H(P) = \text{Æ}_{x \sim P}[-log(P(x)] \tag{36}$$

The final formula can be extended as Equation 38.

$$D_{KL}(P \parallel Q) = \Sigma P(i) log \frac{P(i)}{Q(i)} \tag{37}$$
$$= \int P(x) log \frac{P(x)}{Q(x)} d(x) \tag{38}$$

7.2.6. Correlation Coefficient

Correlation in statistical analysis estimates the relationship of two variables and returns the values between -1 and 1. The positive correlation means that the ranks of both the variables are increasing and the relation between two variables is direct and conversely the negative value shows an indirect relation between two variables.

7.2.7. Fleiss's Kappa

Fleiss's Kappa is one of the methods for measuring of inter-rater reliability between raters. It is a method similar to correlation coefficients that measure the reliability in choice agreement as random. In this thesis, we used Fleiss's Kappa because of the used data set. Fleiss's Kappa is an extension of Cohen's kappa for three raters or more where agreement due to chance is factored out. The value of Kappa can be set between ranges from 0 to 1, where 0 is no agreement, 1 is a perfect agreement, 0.01–0.20 is a slight agreement, 0.21–0.40 is a fair agreement, 0.41– 0.60 is a moderate agreement and 0.61–0.80 is a substantial agreement.

7.2.8. Kendall's Tau Agreement

Kendall's Tau is described as a common measurement of reliability between columns of ranked data. There are different versions of Kendall's Tau including Tau-A, Tau-B, and Tau-C. Tau-A and B are used for square tables while Tau-C is used for rectangular tables. The obtained value of the Tau can be a value between 0 for no relationship and 1 for a perfect relationship. In this thesis, we used Tau-B as a built-in package because of the nature of our data set. Tau-B also can be calculated using Equation 39 as following.

$$Kendall'sTau = \frac{C - D}{C + D} \tag{39}$$

Where the C and the D are the number of concordant and discordant pairs.

7.3. Evaluation Results

Using the evaluation metrics and methodology, the performance of selective personalization and group profiles are evaluated. Queries are sorted according to the potential for personalization metrics and personalization is selectively applied to queries above a threshold.

Because our primary purpose is to improve the re-ranking process using personalization, we measure the Personalization Gain (P-gain) metric introduced by Harvey et al. [4] to compare the number of times the personalization algorithm improves the ranking with the number of times it worsens it.

$$P - gain = \frac{\sum_i^Q 1_{\Delta r(d_i,q_i)<0} - 1_{\Delta r(d_i,q_i)>0}}{\sum_i^Q 1_{\Delta r(d_i,q_i)<0} + 1_{\Delta r(d_i,q_i)>0}} \tag{40}$$

Where Q denotes the number of queries, 1_A is an indicator function that equals one whenever A is true and zero otherwise, and $\Delta r(d, q)$ denotes the change in the rank position of document d for query q resulting from personalization. It can be more simply expressed as the following:

$$P - gain = \frac{\#better - \#worse}{\#better + \#worse} \tag{41}$$

Equation 41 indicates an overall change in the re-ranking process where zero value indicates no change. The positive and negative values indicate an improvement or degradation in performance.

7.3.1. Number of Topics

The number of topics used for LDA is an important parameter. The relationship between MRR and this parameter is investigated in a small development set. Parameters of the LDA model are trained using the training corpus [17]. Figure 7.1. shows the MRR for different topic numbers ranging from 10 topics to 100 for two datasets. The results indicate that using 40 and 30 topics yields the best results in the AOL and TREC2014 dataset.

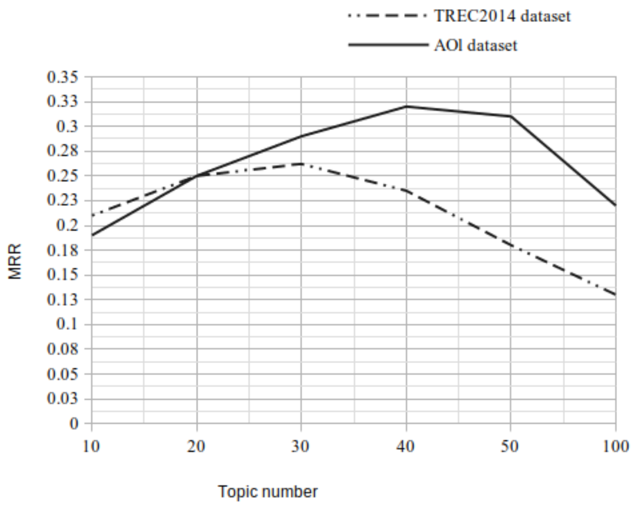

Figure 7.1. The changes in MRR with different topic numbers using the LDA model in the AOL dataset compare with SessionTREC2014 dataset.

7.3.2. Effect of Selective Personalization

The results of the ranking experiments for the models non-personalized $NonPTM(d, q)$ and personalized $PTM(d, q, u)$ are presented in Table 7.2. The results show that there is

[17] Gensim library is used for the LDA estimation https://radimrehurek.com/gensim/

less improvement in the personalized model than in the non-personalized model. The P-gain statistic shows that, on average, the personalized model is improving upon the non-personalized in 3.58% of cases.

	S@1	S@10	MRR@10	P-gain
NonPTM	0.205	0.371	0.267	-
PTM	0.206	0.387	0.272	0.0358

Table 7.2. Ranking performance of PTM and NonPTM methods over all queries. $\lambda = 0.175$

Further inspection revealed that the click entropy of the queries can be used as a significant factor to estimate personalization. In high click entropy queries, the PTM model makes more improvement than low click entropy queries. For these queries the personalized model is able to deliver much better results in comparison to the non-personalized model, registering an improvement with $P - gain$ around 17.97%. In fact the difference in performance over all metrics is significant $(p - value \ll 0.01)$. The improvements are particularly noticeable in the lower ranks resulting in a considerable increase in $S@1$ and MRR.

Besides, we present the experiment in Table 7.3. to investigate the effect of query frequency on the personalization model.

	Low frequency			High frequency		
	S@1	S@10	MRR@10	S@1	S@10	MRR@10
NonPTM	0.192	0.358	0.250	0.213	0.389	0.284
PTM	0.275	0.461	0.362	0.149	0.317	0.216

Table 7.3. Ranking performance of PTM and NonPTM methods on low and high frequency queries (Threshold = 100).

To investigate the importance of selective personalization, the potential for personalization metrics are used to predict the query's potential and they are normalized using the maximum value. Then, for a threshold ξ if the potential is below this value it is ranked with the topic model-based ranking algorithm $NonPTM(d, q)$, otherwise, it is ranked with personalized $PTM(d, q, u)$. A more accurate personalization metric is expected to yield better performance gains with selective personalization as it can identify queries more suitable for personalization.

The result in chapter 6.3.2. on PTM and $NonPTM$ methods show ranking based on topical user profile does not make a large improvement in ranking. So, there is more need for investigation of the results. Based on the obtained results in paper [6], we expect more improvement in the case of short queries and high click entropy in the personalized model. In the personalized methods, we consider factors like click entropy, topic entropy, and unified topic user entropy calculated to reach more improvements. So, we begin to consider known factors such as click entropy, topic entropy calculated for queries in the previous chapter to reach more improvements.

Click Entropy for Estimating Personalization

To start, we calculated click entropy for all queries[18] in our data set and separated queries with low and high click entropy. The result shows that for high click entropy queries, ranking using topical user profile can make more improvement than queries with low click entropy. This is a normal result because as we mentioned before based on some research, click entropy is a good indicator for estimating personalization. Thus, for high click entropy queries using a personalized LDA model or PTM can be useful. Then we divided click entropy in the different ranges from 0 to 1 and listed the result for queries in Table 7.4.

Since the experiments show the MRR for queries with click entropy bigger than 0.6 has the most improvement. So, we tried to use these combinations to present a hybrid model of personalized and non-personalized methods.

The hybrid model uses click entropy as a metric to separate when personalization can be applied. Based on the result in Table 7.4., the hybrid topic model achieves the best performance when click entropy is bigger than 0.6 for personalized and click entropy is less than 0.6 for non-personalized approaches.

But this solution has a problem that for new queries in test data set or queries without history does not provide any solution $(p(d|q) = 0)$. With the further investigation in the data set, this has been seen that around 88% of queries have history and around 11% don't have. So, there

[18]The values are detailed in Figure 0.1., Appendix A.

is a need to inspect the potential for personalization among them and make a ranking method to reach a performance when click entropy is not available. Furthermore, a large percentage of queries in data set are common or popular queries with too low click entropy means that the clicked results for all people are the same. For them, ranking based on the prior click obtains a better result.

	Click Entropy					
	$\xi > 0.0$	$\xi > 0.2$	$\xi > 0.4$	$\xi > 0.6$	$\xi > 0.8$	$\xi > 1.0$
S@1	0.206	0.227	0.309	0.353	0.331	0.205
S@10	0.387	0.413	0.454	0.498	0.477	0.371
MRR	0.272	0.298	0.354	0.416	0.382	0.267

Table 7.4. Ranking performance on different combination of PTM and Non-PTM methods for different range of click entropy.

Topic Entropy for Estimating Potential

In addition to click entropy, in a similar method, we tried a different range of topic entropy between [0,1] to present a hybrid model of personalized and non-personalized methods. The result is showed in Table 7.5. Based on the result in Table 7.5., the Hybrid topic model achieves more accurate around 7% than click entropy. It means that using topic entropy as a personalization metric achieves a better result than using click entropy. It is probably due to using a topic distribution of documents and accompanying the content of documents.

This solution has also drawbacks. For new queries in test data set or queries without history does not provide any solution. For example, when the user profile includes topics in the field of "computer software" and user searches for a new query such as "robotic". Here we need a mechanism to decision about to provide the ranking result as unpersonalized or personalized (using user profile). We can use the relations between the topic distribution of user profile and query as a new metric to estimate the potential for personalization in a query.

With the further investigation in the data set, there is a need to inspect the potential for personalization among queries. It makes a need for a ranking method to reach a performance when click entropy and topic entropy are not available. Here we discuss providing a solution to the problem. We estimated the potential for personalization for queries using the presented metric (unified topic user entropy (UTUE)) in Chapter 4.

Using the metric presented we can present a solution for both queries with and without history. Based on the presented metric we provide a hybrid model by combining PTM and $NonPTM$ methods and use the provided metric as a weighting approach to estimate the potential for personalization.

	Topic Entropy					
	$\xi > 0.0$	$\xi > 0.2$	$\xi > 0.4$	$\xi > 0.6$	$\xi > 0.8$	$\xi > 1.0$
S@1	0.206	0.298	0.359	0.413	0.328	0.205
S@10	0.387	0.482	0.542	0.572	0.496	0.371
MRR	0.272	0.378	0.420	0.481	0.385	0.267

Table 7.5. Ranking performance on different combination of PTM and Non-PTM methods for different range of Topic entropy.

Unified Topic User Entropy(UTUE) for Estimating Potential

Tables 7.6. and 7.7. report the MRR, $S@1$, $S@10$ and $nDCG@10$ scores for the three potential for personalization metrics in AOL and Session Track 2014 datasets. A complete comparison of selective personalization using different potential for personalization metrics in AOL, session TREC 2013 and 2014 datase is depicted in Figures 0.1., 0.2., 0.3., 0.4., 0.5., and 0.6., Appendix B. In Table 7.6. and 7.7. the first row represents the ranking score when using only $NonPTM(d, q)$, which is no personalization. The last row shows the result when all queries are re-ranked using $PTM(d, q, u)$. Naturally, these two cases are independent of the potential metric used and are common for all three metrics. When we consider the results of UTUE, it is evident that it achieves a higher score for all different thresholds. It indicates that it assigns a more accurate prediction for personalization, and the queries with lower UTUE score does not benefit from personalization. A similar result is observed between Topic Entropy and Click Entropy, confirming the experiments in Yano et al. [3], Topic entropy performs better than Click entropy.

The results of UTUE for $\xi > 0.6$ achieves the highest-ranking scores for all measures. It indicates that using personalization only for queries with a potential higher than 0.6 is a better

ξ	Click Entropy			Topic Entropy			UTUE		
	S@1	S@10	MRR	S@1	S@10	MRR	S@1	S@10	MRR
None	0.205	0.371	0.267	0.205	0.371	0.267	0.205	0.371	0.267
$\xi > 0.8$	0.331	0.477	0.382	0.328	0.496	0.385	0.384	0.560	0.445
$\xi > 0.6$	0.353	0.498	0.416	0.413	0.572	0.481	**0.462**	**0.620**	**0.536**
$\xi > 0.4$	0.309	0.454	0.354	0.359	0.542	0.420	0.408	0.601	0.478
$\xi > 0.2$	0.227	0.413	0.298	0.298	0.482	0.378	0.357	0.542	0.431
All	0.206	0.387	0.272	0.206	0.387	0.272	0.206	0.387	0.272

Table 7.6. Comparison of selective personalization using different potential for personalization metrics in AOL dataset. Personalization PTM is applied only to queries with potential $> \xi$.

ξ	Click Entropy				Topic Entropy				UTUE			
	S@1	S@10	MRR	nDCG@10	S@1	S@10	MRR	nDCG@10	S@1	S@10	MRR	nDCG@10
None	0.163	0.328	0.231	0.259	0.163	0.328	0.231	0.259	0.163	0.328	0.231	0.259
$\xi > 0.8$	0.268	0.418	0.327	0.340	0.275	0.461	0.357	0.403	0.322	0.517	0.426	0.452
$\xi > 0.6$	0.307	0.465	0.372	0.389	0.376	0.539	0.440	0.491	**0.419**	**0.588**	**0.493**	**0.527**
$\xi > 0.4$	0.272	0.420	0.339	0.351	0.332	0.507	0.384	0.438	0.356	0.540	0.442	0.469
$\xi > 0.2$	0.203	0.383	0.285	0.302	0.256	0.451	0.359	0.407	0.304	0.507	0.418	0.440
All	0.171	0.364	0.265	0.298	0.171	0.364	0.265	0.298	0.171	0.364	0.265	0.298

Table 7.7. Comparison of selective personalization using different potential for personalization metrics in sessionTREC2014 dataset.

strategy than using other thresholds. When considering the difference between applying personalization to all of the queries and selective personalization with $\xi > 0.6$, the performance gain for MRR is as high as 0.264 in the AOL dataset and 0.228 in TREC 2014.

7.3.3. User Topical Profile versus Group Topical Profile

To solve the sparsity problem in some user profiles, we propose group profiles. These group profiles can be enriched using a group of user-profiles and then be utilized to re-rank the list of documents.

To test our hypothesis that group profiles resolve the sparsity problem, we compared the selective personalization effectiveness of user-based $PTM(d, q, u)$ and group-based $GPTM(d, q, u)$. Using the threshold $\xi = 0.6$, the two ranking methods are compared. An important parameter defined by Vu et al. [172] is the temporal decaying model for the documents clicked by the users. To take into account the difference between a user or group profile formed of short and long-term user interactions, two separate experiments are performed.

The first short-term experiment uses the document clicks performed in a month by the users, while the long-term use all the clicked documents. To investigate the relationship between the number of clusters and MRR, different number of clusters are evaluated ($k \in$ {10, 20, 30, 40 ,50 ,100}) in Figure 0.6. As depicted in Figure 7.2. the best result is obtained with $k = 30$.

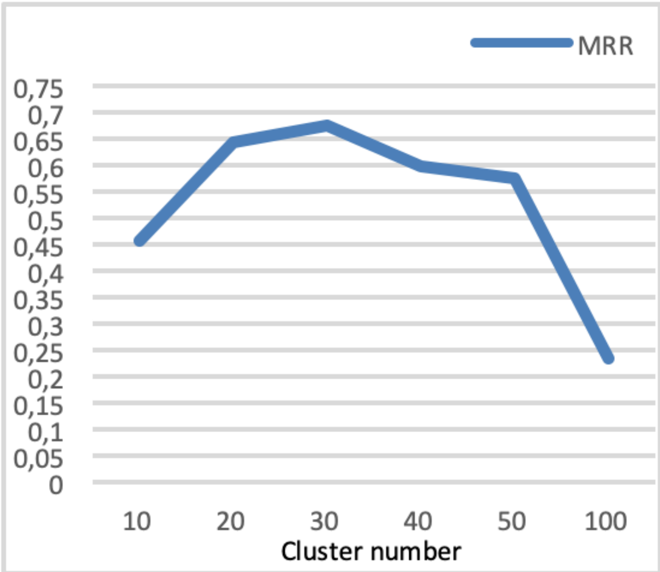

Figure 7.2. The changes in MRR in GPTM model in different cluster numbers.

Figure 7.3. shows the MRR based performance comparison for the two models PTM and $GPTM$. As can be observed, using group profiles improves MRR by 0.09 in long-term profiles, while this value is 0.08 for short-term profiles. So, using group profiles instead of user-profiles improves both cases. As expected, the short-term user profile is more effective than the long-term.

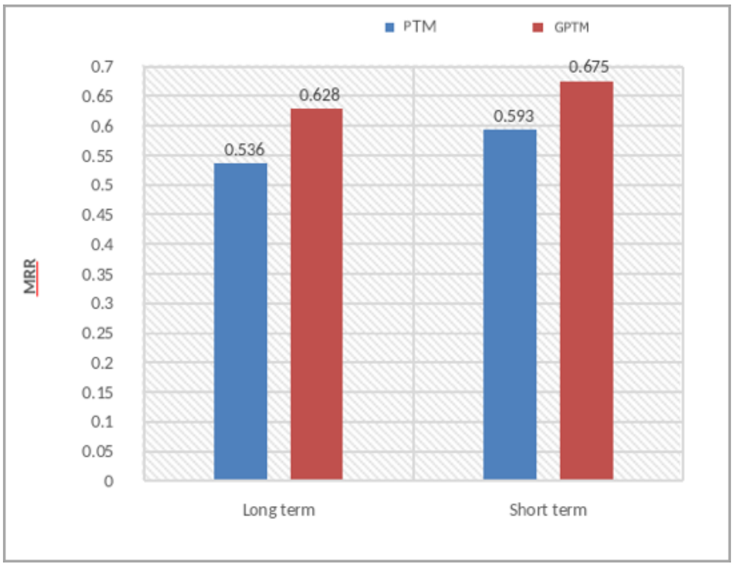

Figure 7.3. Ranking performance of the two models PTM and GPTM on queries.

8. CONCLUSION AND THOUGHTS FOR THE FUTURE

We have believed that the personalization process can be improved using estimating the potential for personalization in queries. There are known metrics based on the previous research to estimate the potential for personalization however they have limitations when click entropy and topic entropy are not available. In this thesis, a selective personalization strategy is proposed. As one important stumbling block, the potential for personalization of new or low-frequency terms is not handled by the state-of-the-art metrics. We presented a metric to handle such new queries, which make-up an important portion of the queries in a search engine's log.

When compared to the method proposed by Yano et al. [3], our proposed potential for personalization metric is defined in terms of the latent topic models, rather than relying solely on the query history directly. This allows the $UTUE$ to generalize better to rare queries as well as new queries that are not issued previously as it is. Using the topic models, these queries are modeled using similar queries in a more flexible way.

Furthermore, we show that selective personalization using a combination of $UTUE$ and topic entropy improves personalization effectiveness. Handling low-frequency queries with $UTUE$ and reverting to topic entropy for the other queries, a better selective personalization strategy is proposed. Our results indicate a 4-5$\%$ improvement with only this strategy. This proves that handling low frequency queries better is an important subtask for such selective personalization systems.

In the other part of the research, to investigate keyphrase-based user profiles in the personalized web search, it is considered how integration between keyphrase extraction and personalization by the state-of-the-art approaches. The personalization methods are created using supervised and unsupervised keyphrase extraction methods. For evaluation of the model, keyphrase-based user profiles using the re-ranking algorithms are applied using different datasets. The personalized models based on the supervised keyphrase extraction approaches obtained more accuracy around 7% than unsupervised approaches. In our experiments, among supervised approaches, both feature-based methods and graph-based methods resulted in the same improvement average by 26% in $nDCG$ score for long-term compared to 36% for session-term profiles created using Session Track 2014 dataset.

In the last part of the research, we searched for the answering of the question "personalized topical profile or group topical profile". In this part, we evaluated two models PTM and $GPTM$ on queries as short and long term. The mean reciprocal rank obtained by the $Short-GTM$ exceeded %67. Based on the results, it can be inferred that group profiling reaches an improvement of around %62 for long and %67 for short-term profiles. Noting a similar sparsity problem in user profiles based on topic models, rather than depending solely on the user performing the query and consolidated profile formed of similar users improves personalization. The proposed group profiles improve the MRR of the queries by 9% and 8% respectively for the long-term and short-term profiles. The topic model-based search system achieves a 67% MRR score. To the best of our knowledge, grouping users by their profiles built using topic models is a novel method. We presented these results and experiments in a published paper by "TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES" [282].

Appendix A

The Changes of Query Percentage to Click Entropy and Query Length on the AOL Query Log

As we mentioned before one of the effective factors in the performance of personalization is click entropy. Some research like [2] has discussed on click entropy as a factor in the quality of the search results. A small value of click entropy indicates a small range of clicked web pages for a query. Therefore we are aimed to explore the click entropy distribution on query percentage. Figure 0.1. depicted the distribution of click entropy in the AOL query log. The result shows 45.05% of queries have a click entropy between 0 and 0.5 and 28.14% queries have from 0.5 to 1. This value is 14.05% for queries with click entropy between 1 and 1.5 and 7.95% for queries between 1.5 and 2. Only 4.81% queries have a click entropy (≥ 2).

The other factor that has been discussed in various studies is the length of the query in the character and the word. It may be showed the specific information need in longer queries, while in short queries the less information is indicated. In Figure 0.2., we depicts the distribution of the query length in the AOL data set. As a statistic, the most queries have less than 3 words about 86% while the number of queries containing 4 and 5 words are 9.7% and 4.3%, respectively.

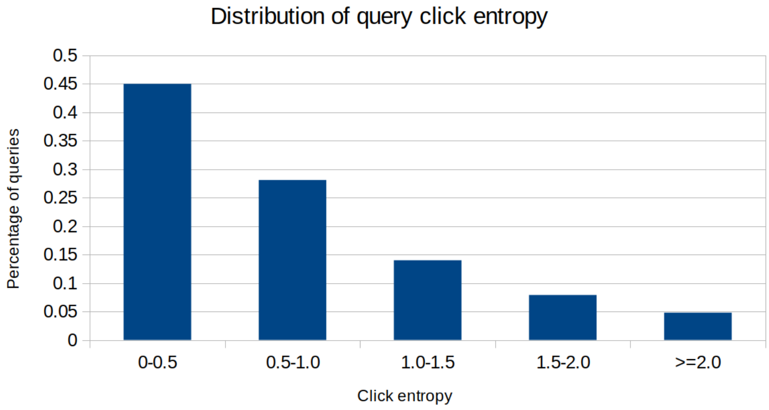

Figure 0.1. Distribution of query click entropy.

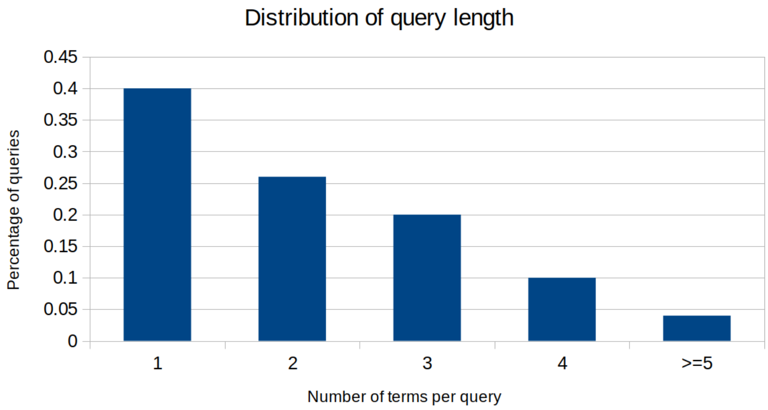

Figure 0.2. The percentage of the query length with terms.

Since click entropy and topic entropy are known used metrics to estimate potential for personalization, we consider the relationship between these metrics. Therefore the correlation between click entropy, topic entropy and query length are calculated in Table 0.1.

	Click Entropy	Topic Entropy	Query Length in words
Click Entropy	1.0	0.771	0.233
Query Length in words	0.233	0.382	1.0
Topic Entropy	0.771	1.0	0.382

Table 0.1. Correlation between click entropy, topic entropy and query length for the queries.

The results in Table 0.1. shows a high correlation between click entropy and topic entropy around 77% while a low correlation between click or topic entropy and query length(in words).

Appendix B

Effect of Selective Personalization

Figures 0.1., 0.2., 0.3., 0.4., 0.5., and 0.6. report the MRR, and $nDCG@10$ scores for the four potential for personalization metrics in AOL and Session Tracks 2013 and 2014 datasets. In all figures there are two ranges $[0.0 - 1.0]$ and $[1.0 - 1.0]$. The first one ($[0.0 - 1.0]$ range) shows the MRR result when all queries are re-ranked using $PTM(d, q, u)$ and the other ($[1.0 - 1.0]$ range) represents the ranking score when using only $NonPTM(d, q)$, which is no personalization. Naturally, these two cases are independent of the potential metric used and are common for all four metrics. When we consider the results of UTUE, it is evident that it achieves a higher score for all different thresholds. This indicates that it assigns a more accurate prediction for personalization, and the queries with lower UTUE score does not benefit from personalization. A similar result is observed between Topic Entropy and Click Entropy, confirming the experiments in Yano et al. [3], Topic entropy performs better than Click entropy.

The results of UTUE for $[0.6 - 1.0]$ achieves the highest-ranking scores for all measures. This indicates that using personalization only for queries with a potential higher than 0.6 is a better strategy than using other thresholds. When considering the difference between applying personalization to all of the queries and combination personalization model with $[0.6 - 1.0]$, the performance gain for MRR is as high as 0.264 in the AOL dataset, 0.224 in TREC 2014 and 0.241 in TREC 2013.

Appendix B. *Comparison of Selective Personalization in AOL, Session-TREC2013 and 2014 Datasets*

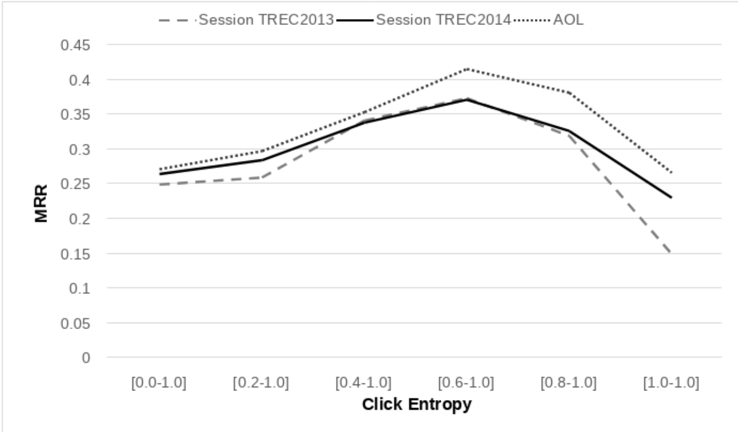

Figure 0.1. The Changes in MRR with different ranges of click entropy using selective personalization.

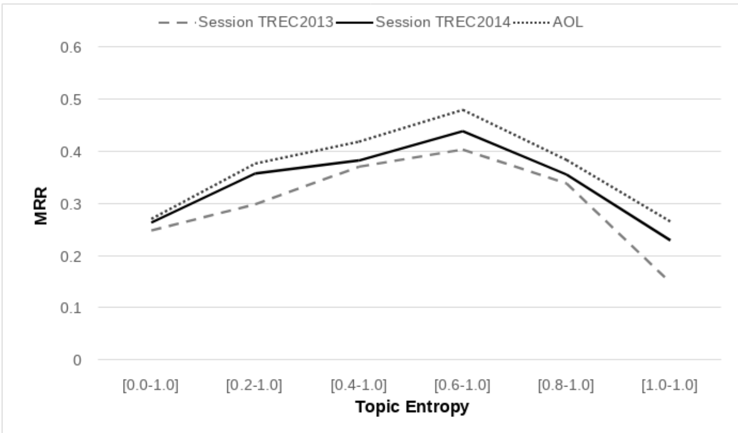

Figure 0.2. The Changes in MRR with different ranges of topic entropy using selective personalization.

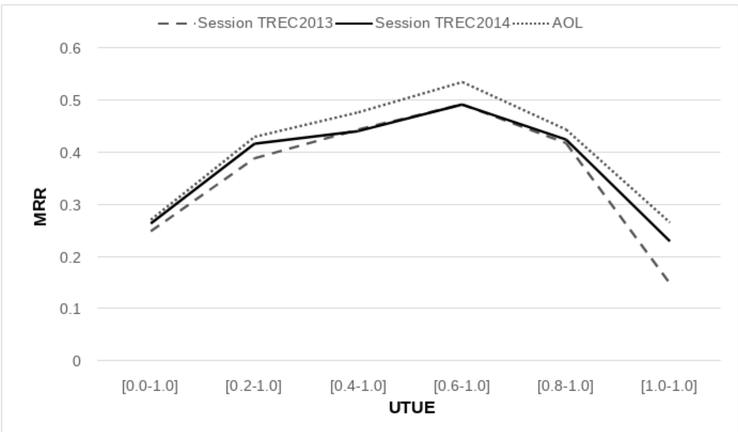

Figure 0.3. The Changes in MRR with different ranges of UTUE using selective personalization.

Appendix B. *Comparison of Selective Personalization in AOL, Session-TREC2013 and 2014 Datasets*

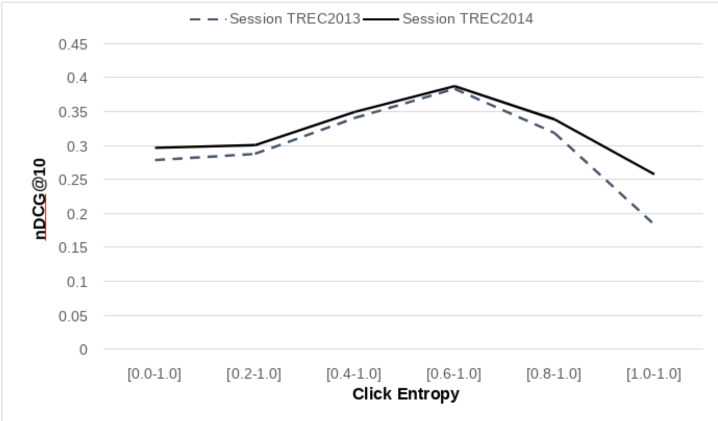

Figure 0.4. The Changes in $nDCG@10$ with different ranges of click entropy using selective personalization.

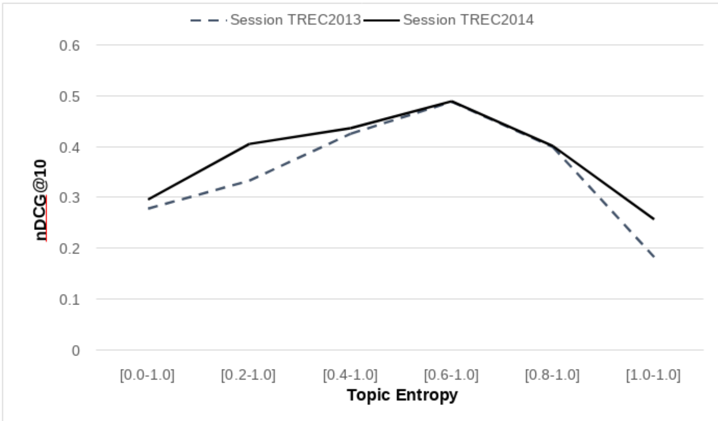

Figure 0.5. The Changes in $nDCG@10$ with different ranges of topic entropy using selective personalization.

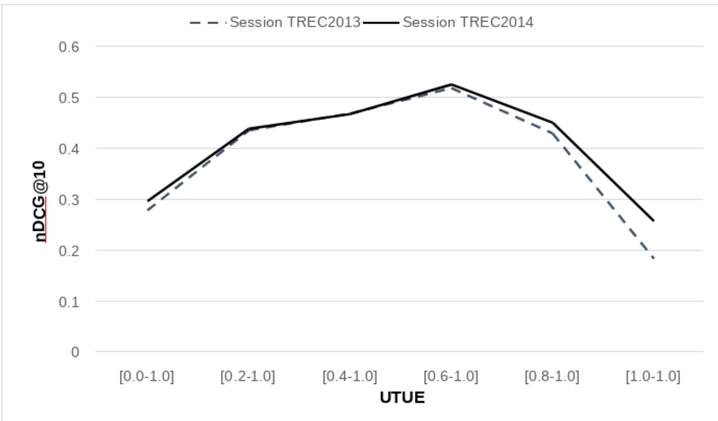

Figure 0.6. The Changes in $nDCG@10$ with different ranges of UTUE using selective personalization.

REFERENCES

[1] Jaime Teevan, Susan Dumais, and Eric Horvitz. Beyond the commons: Investigating the value of personalizing web search. In *Proceedings of the Workshop on New Technologies for Personalized Information Access (PIA)*, pages 84–92. **2005**.

[2] Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. A large-scale evaluation and analysis of personalized search strategies. In *Proceedings of the 16th international conference on World Wide Web*, pages 581–590. ACM, **2007**.

[3] Yuki Yano, Yukihiro Tagami, and Akira Tajima. Quantifying query ambiguity with topic distributions. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1877–1880. ACM, **2016**.

[4] Morgan Harvey, Fabio Crestani, and Mark J Carman. Building user profiles from topic models for personalised search. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 2309–2314. ACM, **2013**.

[5] Mark J Carman, Fabio Crestani, Morgan Harvey, and Mark Baillie. Towards query log based personalization using topic models. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1849–1852. ACM, **2010**.

[6] Jaime Teevan, Susan Dumais, and Eric Horvitz. Potential for personalization. In *ACM Transactions on Computer-Human Interaction (TOCHI) TOCHI*, pages 1–31. ACM, **2010**.

[7] Jaime Teevan, Susan T Dumais, and Daniel J Liebling. To personalize or not to personalize: modeling queries with variation in user intent. In *Proceedings of*

the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pages 163–170. ACM, **2008**.

[8] Steve Cronen-Townsend, Yun Zhou, and Bruce Croft. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM, **2002**.

[9] Yu Wang and Eugene Agichtein. Query ambiguity revisited: clickthrough measures for distinguishing informational and ambiguous queries. In *HLT '10 Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 361–364. ACM, **2010**.

[10] Ruihua Song, Zhenxiao Luo, Ji-Rong Wen, Yong Yu, and Hsiao-Wuen Hon. Identifying ambiguous queries in web search. In *Proceedings of the 16th international conference on World Wide Web*, pages 1169–1170. ACM, **2007**.

[11] Mohammad-Reza Tazari, Matthias Grimm, and Matthias Finke. Modeling user context. In *Proceedings of the 10th International on Conference on Human-Computer Interaction*. **2003**.

[12] Matthias Schneider-Hufschmidt, Uwe Malinowski, and Thomas Kühme. Adaptive user interfaces: Principles and practices. In *Adaptive User Interfaces: Principles and Practices*. ELSEVIER, **1993**.

[13] Bernd Gutkauf, Stefanie Thies, and Gitta Domik. A user-adaptive chart editing system based on user modeling and critiquing. In *Proceedings of the Sixth International Conference on User Modeling*. **1997**.

[14] Pat Langley. User modeling in adaptive interfaces. In *Proceedings of the seventh international conference on User modeling*, pages 357–370. ACM, **1999**.

[15] Hai Zhuge. China's e-science knowledge grid environment. *IEEE Intelligent Systems*, 19(1):13–17, **2004**.

[16] Matthew Montebello, Gray W.A., and Steve Hurley. A personable evolvable advisor for www knowledge-based systems. In *Proceedings of 1998 Intl. Database Engineering and Application Symposium*, pages 224–233. ACM, **1998**.

[17] Chunyan Liang. User profile for personalized web search. In *International Conference on Fuzzy Systems and Knowledge Discovery*, pages 1847–1850. **2011**.

[18] Jie Tang, Limin Yao, Duo Zhang, and Zhang Jing. A combination approach to web user profiling. *ACM Transactions on Knowledge Discovery from Data*, 5(1), **2010**.

[19] Marko Balabanovic and Shoham Yoav. Learning information retrieval agents: Experiments with automated web browsing. In *On-line Working Notes of the AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments*. **1995**.

[20] Eric J. Horvitz, John S. Breese, David Heckerman, David Hovel, and Koos Rommelse. The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 256–265. **1998**.

[21] Rich Elaine. Users are individuals: individualizing user models. *International Journal of Human-Computer Studies*, 51(2):323–338, **1999**.

[22] Phivos Mylonas, David Vallet, Pablo Castells, and Miriam Fernandez. Personalized information retrieval based on context and ontological knowledge. *The Knowledge Engineering Review*, 23(1):73–100, **2008**.

[23] Robert Dinoff, Richard Hull, Bharat Kumar, Daniel Lieuwen, and Paulo Santos. Learning and managing user context in personalized communications services. In *Proceedings of the international workshop in conjunction with AVI 2006 on Context in advanced interfaces*, pages 33–36. **2006**.

[24] Danny Raz, Arto Juhola, Joan Serrat-Fernandez, and Alex Galis. Fast and efficient context-aware services. In *Fast and Efficient Context-Aware Services*. John Wiley & Sons, Ltd, **2006**.

[25] Dwi H. Widyantoro, Thomas R. Ioerger, and John Yen. Learning user interest dynamics with three-descriptor representation. *Journal of the American Society for Information Science and Technology*, 52(3):212–225, **2001**.

[26] Keiichiro Hoashi, Kazunori Matsumoto, Naomi Inoue, and Kazuo Hashimoto. Document filtering method using non-relevant information profile. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 176–183. ACM SIGIR, **2000**.

[27] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. **2002**.

[28] Ahmed Nazer, Tarek Helmy, and Muhammed Al-Mulhem. User's profile ontology-based semantic framework for personalized food and nutrition recommendation. *Procedia Computer Science*, 32:101–108, **2014**.

[29] Rich Elaine. Stereotypes and user modeling. In *User Models in Dialog Systems, Springer-Verlag*, pages 35–51. **1989**.

[30] Vasudeva Varma and Nithin Kumar. Generating simulated relevance feedback: A prognostic search approach. In *COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 597–604. **2010**.

[31] Michael Pazzani, Jack Muramatsu, and Daniel Billsus. Syskill & webert: Identifying interesting web sites. In *Proceedings of the 13th National Conference on Artificial Intelligence Portland*, pages 54–61. **1996**.

[32] Ah-Hwee Tan and C. Teo. Learning user profiles for personalized information dissemination. In *Proceedings of International Joint Conference on Neural Network*, pages 183–188. **1998**.

[33] Jude Shavlik and Tina Eliassi-Rad. Intelligent agents for web-based tasks: An advice-taking approach. In *Working Notes of the AAAI/ICML-98 Workshop on Learning for text categorization*. **1999**.

[34] Jude Shavlik, Susan Calcari, Tina Eliassi-Rad, and Jack Solock. An instructable, adaptive interface for discovering and monitoring information on the world wide web. In *Proceedings of the 1999 International Conference on Intelligent User Interfaces*, pages 157–160. **1999**.

[35] Ivan M. Koychev and Ingo Schwab. Adaptation to drifting user's interests. In *Proceedings of ECML2000 Workshop: Machine Learning in New Information Age*. **2000**.

[36] John M. Carroll and Mary B. Rosson. The paradox of the active user," interfacing thought: Cognitive aspects of human-computer interaction. In *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*, pages 80–111. **1998**.

[37] Jinmook Kim and Douglas Oard. Using implicit feedback for user modeling in internet and intranet searching. In *University of Maryland CLIS Technical Report*. **2000**.

[38] Diane Kelly and Jaime Teevan. Implicit feedback for inferring user preference: a bibliography. *ACM SIGIR Forum*, 37(2):18–28, **2003**.

[39] Mark Claypool, Phong Le, Makoto Waseda, and David Brown. Implicit interest indicators. In *Proceedings of the 6th international conference on Intelligent user interfaces*, pages 33–40. **2001**.

[40] Douglas Oard and Jinmook Kim. Implicit feedback for recommender systems. In *Proceedings of the AAAI Workshop on Recommender Systems*, pages 81–83. **1998**.

[41] Faten Khalil, Jiuyong Li, and Hua Wang. Integrating recommendation models for improved web page prediction accuracy. In *Proceedings of the thirty-first Australasian conference on Computer science*, pages 91–100. **2008**.

[42] Sule Gündüz and Tamer Özsu. Recommendation models for user accesses to web pages. In *Proceedings of the 2003 joint international conference on Artificial neural networks and neural information processing*, pages 1003–1010. **2003**.

[43] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings 13th International Conference on World Wide Web*, pages 675–684. **2004**.

[44] Cuncun Wei, Chongben Huang, and Hengsong Tan. A personalized model for ontology-driven user profiles mining. In *2009 International Symposium on Intelligent Ubiquitous Computing and Education*, pages 484–487. **2009**.

[45] Ann Lieberman. Practices that support teacher development: Transforming conceptions of professional learning. In *Phi Delta Kappan*, pages 591–596. **1995**.

[46] Feng Qiu and Junghoo Cho. Automatic identification of user interest for personalized search. In *Proceedings of the 15th international conference on World Wide Web*. **2006**.

[47] Jaime Teevan, Susan Dumais, and Eric Horvitz. Personalizing search via automated analysis of interests and activities. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 449–456. **2005**.

[48] James Pitkow, Hinrich Schütze, Todd Cass, Rob Cooley, Don Turnbull, Andy Edmonds, Eytan Adar, and Thomas Breuel. Personalized search. *Communications of the ACM*, 45(9):50–55, **2002**.

[49] Geoffrey Webb and Mark Kuzmycz. Feature based modelling: A methodology for producing coherent, consistent, dynamically changing models of agents competencies. In *User Modelling and User-Adapted Interaction*, pages 117–150. Springer, **1996**.

[50] Maria Golemati, Akrivi Katifori, Costas Vassilakis, and Constantin Halatsis. Creating an ontology for the user profile: Method and applications. In *Proceedings of the First International Conference on Research Challenges in Information Science*, pages 407–412. **2007**.

[51] Bark ChiuGeoffrey and Webb Geoffrey. Using decision trees for agent modelling: Improving prediction performance. *User Modeling and User-Adapted Interaction*, 8(1-2):131–152, **1998**.

[52] Robert Guttman, Alexandros Moukas, and Pattie Maes. Agent-mediated electronic commerce: A survey. *The Knowledge Engineering Review*, 13(2):147–159, **1998**.

[53] Danny Poo, Chng Brian, and Goh Jie-Mein. A hybrid approach for user profiling. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, pages 9–13. IEEE, **2003**.

[54] Ralf Klinkenberg and Ingrid Renz. Adaptive information filtering: Learning in the presence of concept drifts. In *Workshop Notes of the ICML/AAAI-98 Workshop on Learning for Text Categorization*, pages 33–40. **1998**.

[55] Wai Lam and Javed Mostafa. Modeling user interest shift using a bayesian approach. *Journal of the American Society for Information Science and Technology*, 52(5):416–429, **2001**.

[56] J. P. McGowan, Nicholas Kushmerick, and Barry Smyth. Who do you want to be today? web personae for personalized information access. In *Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 514–517. Springer, **2002**.

[57] Anton Vedder. Kdd: The challenge to individualism. *Ethics and Information Technology*, 1(4):275–281, **1999**.

[58] Josef Fink and Alfred Kobsa. A review and analysis of commercial user modelling servers for personalization of the world wide web. *User Modeling and User-Adapted Interaction*, 10(2-3):209–249, **2000**.

[59] Dominik Heckmann, Tim Schwartz, Boris Brandherm, Michael Schmitz, and Margeritta Wilamowitz-Moellendorff. Gumo - the general user model ontology. In *International Conference on User Modeling*, pages 428–432. Springer, **2005**.

[60] Ali Ghulam, Noor Shaikh, and Zubair Shaikh. Agent-based user-profiling model for behavior monitoring. In *International Conference on Future Networks*. IEEE, **2009**.

[61] Alexander Pretschner and Susan Gauch. Ontology based personalized search. In *Proceeding of 11th IEEE International Conference on Tools with Artificial Intelligence*, pages 391–398. IEEE, **1999**.

[62] Ugur Çetintemel, Mike Franklin, and Lee Giles. Self-adaptive user profiles for large scale data delivery. In *Proceedings of the 16th International Conference on Data Engineering*, pages 622–636. IEEE, **2000**.

[63] Fang Liu, Clement Yu, and Weiyi Meng. Personalized web search by mapping user queries to categories. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 558–565. **2002**.

[64] Han-joon Kim, Sungjick Lee, Byungjeong Lee, and Sooyong Kang. Building concept network-based user profile for personalized web search. In *9th*

IEEE/ACIS International Conference on Computer and Information Science, pages 567–572. **2010**.

[65] Fang Liu, Clement Yu, and Weiyi Meng. Personalized web search for improving retrieval effectiveness. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):28–40, **2004**.

[66] Lynda Lechani, Mohand Boughanem, and Nesrine Zemirli. Personalized document ranking: exploiting evidence from multiple user interests for profiling and retrieval. *Journal of Digital Information Management*, **2008**.

[67] Susan Gauch, Jason Chaffee, and Alexander Pretschner. Ontology-based user profiles for search and browsing, user modeling and user-adapted interaction. *The Journal of Personalization Research, Special Issue on User Modeling for Web and Hypermedia Information Retrieval*, 1(3-4):219–234, **2003**.

[68] Ahu Siegg, Bamshad Mobasher, and Robin Burke. Web search personalization with ontological user profiles. In *Proceedings of the ACM Conference on information and knowledge management*, pages 525–534. ACM, **2007**.

[69] Francisco Tanudjaja and Lik Mui. Persona: A contextualized and personalized web search. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, page 67. IEEE, **2002**.

[70] Silvia Calegari and Gabriella Pasi. Ontology-based information behaviour to improve web search. *Future Internet*, 2(4):533–558, **2010**.

[71] Mariam Daoud, Lynda Tamine, and Mohand Boughanem. A personalized graph-based document ranking model using a semantic user profile. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 171–182. Springer, **2010**.

[72] Mirco Speretta and Susan Gauch. Miology: a web application for organizing personal domain ontologies. In *Proceedings of the International Conference*

on Information, Process and Knowledge Management, pages 159–161. IEEE, **2009**.

[73] Joana Trajkova and Susan Gauch. Improving ontology-based user profiles. In *Proceedings of the RIAO '04 Coupling approaches, coupling media and coupling languages for information retrieval*, pages 380–390. **2004**.

[74] Susan Gauch, Mirco Speretta, Aravind Chandramouli, and Alessandro Micarelli. User profiles for personalized information access. In *The Adaptive Web 2007*, pages 54–89. Springer, **2007**.

[75] Michael Shepherd, Carolyn Watters, and Jack Duffy. Browsing and keyword-based profiles: a cautionary tale. In *Proceedings of the 34th Hawaii International Conference on System Sciences*, pages 1365–1373. IEEE, **2001**.

[76] Michal Barla and Mária Bielikova. Ordinary web pages as a source for metadata acquisition for open corpus user modeling. In *IADIS International Conference WWW/Internet 2010*, pages 227–233. **2010**.

[77] Paul Baclace. Personal information intake filtering. In *Proceedings of Bellcore Workshop on High-Performance Information Filtering*. **1991**.

[78] Gerard Salton, Andrew Wong, and Chungshu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, **1975**.

[79] Gerard Salton and Michael McGill. *Introduction to Modern Information Retrieval*. Ph.D. thesis, McGraw-Hill, **1983**.

[80] Mohammed Nazim, Duong Trong, and Jo Geun. Contextual information search based on ontological user profile. In *International Conference on Computational Collective Intelligence*, pages 490–500. Springer, **2010**.

[81] Duong Trong, Mohammed Nazim, Delong Li, and Jo Geun. A collaborative ontology-based user profiles system. In *International Conference on Computational Collective Intelligence*, pages 540–552. Springer, **2009**.

[82] Samira Karimi mansoub and Rahem Abri. Improvement of semantic search results with providing an updatable dynamic user model. *International Journal of Computer Applications*, 155(4):7–14, **2016**.

[83] Mariam Daoud, Lynda Tamine-Lechani, and Mohand Boughanem. A session based personalized search using an ontological user profile. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 1732–1736. ACM, **2009**.

[84] Nikolaos Nanas, Manolis Vavalis, and Lefteris Kellis. Immune learning in a dynamic information environment. In *Proceedings of the 8th international Conference on Artificial Immune Systems*, pages 192–205. **2009**.

[85] James Rucker and Marcos Polanco. Siteseer: Personalized navigation for the web. *Communications of the ACM*, 40(3):73–75, **1997**.

[86] Humphrey Sorensen and M Mc Elligott. Psun: A profiling system for usenet news. In *Proceedings of CIKM'95 Workshop on Intelligent Information Agents Workshop*. **1995**.

[87] Anna Stefani and Carlo Strappavara. Personalizing access to web sites: The siteif project. In *Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia HYPERTEXT' 98*. **1998**.

[88] Marko Turpeinen. Agents in delivering personalized content based on semantic metadata. In *Proceedings of 1999 AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace*, pages 84–93. **1999**.

[89] Hyoung Kim and Philip Chan. Learning implicit user interest hierarchy for context in personalization. In *Proceedings of the 8th international conference on Intelligent user interfaces*, pages 101–108. **2003**.

[90] Daniela Godoy and Analía Amandi. Modeling user interests by conceptual clustering. *Information Systems*, 31(4):247–265, **2006**.

[91] Yihong Li and Fang Li. Modeling hierarchical user interests based on hownet and concept mapping. In *Proceedings of the 2008 Fourth International Conference on Semantics, Knowledge and Grid*, pages 157–164. **2008**.

[92] Xiaochuan Ni, Xiaoyuan Wu, and Yong Yu. Automatic identification of chinese weblogger's interests based on text classification. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 247–253. **2006**.

[93] Andrew Mccallum and Kamal Nigam. A comparison of event models for naïve byaes text classification. In *Proceedings of the AAAI-98 Workshop on "Learning for Text Categorization*, pages 41–48. **1998**.

[94] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the ECML-98*, pages 137–142. **1998**.

[95] Thorsten Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *Proceedings of the ICML-97*, pages 143–151. **1997**.

[96] Chien Chin Chen, Meng Chang Chen, and Yeali Sun. Pva: A self-adaptive personal view agent system. *Journal of Intelligent Information Systems*, 18(2):173–194, **2002**.

[97] Ahu Sieg, Bamshad Mobasher, and Robin Burke. Inferring users information context: Integrating user profiles and concept hierarchies. In *2004 Meeting of the International Federation of Classification Societies*. **2004**.

[98] Ni Na and Li Yaodong. User interests modeling in online forums. In *International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM*, pages 708–709. **2012**.

[99] Peng Xueping, Zhendong Niu, Sheng Huang, and Yumin Zhao. Personalized web search using clickthrough data and web page rating. *Journal of Computers*, 7(10):2578–2584, **2012**.

[100] David Vallet, Miriam Fernández, and Pablo Castells. An ontology-based information retrieval model. In *Proceedings of the 2nd European Semantic Web Conference (ESWC)*, pages 455–470. **2005**.

[101] Ahu Sieg, Bamshad Mobasher, and Robin Burke. Learning ontology-based user profiles: A semantic. approach to personalized web search. *IEEE Intelligent Informatics*, 8(1), **2007**.

[102] Hyunjang Kong, Myunggwon Hwang, and Pankoo Kim. Design of automatic ontology building system about the specific domain knowledge. In *2006 8th International Conference Advanced Communication Technology*, pages 20–22. **2006**.

[103] Alexander Maedche and Steffen Staab. Mining ontologies from text. In *EKAW 2000*, pages 189–202. **2000**.

[104] Christiaan Fluit, Marta Sabou, and Frank van Harmelen. Ontology-based information visualisation. In *Visualising the Semantic Web, Springer Verlag 2002*. **2002**.

[105] Natalya Fridman Noy and Carole D. Hafner. The state of the art in ontology design, a survey and comparative review. *AI Magazine*, 18(3):53–74, **1997**.

[106] Matteo Cristani and Roberta Cuel. A survey on ontology creation methodologies. *International Journal on Semantic Web and Information Systems*, 1(2):49–69, **2005**.

[107] Ngoc Thanh Nguyen. Inconsistency of knowledge and collective intelligence. *Cybernetics and Systems*, 39(6):542–562, **2008**.

[108] Ngoc Thanh Nguyen. A method for ontology conflict resolution and integration on relation level. *Cybernetics and Systems*, 38(8), **2007**.

[109] Steve Lawrence. Context in web search. *IEEE Data Engineering Bulletin*, 23(3):25–32, **2000**.

[110] Czeslaw Danilowicz and Ngoc Thanh Nguyen. Consensus-based partitions in the space of ordered partitions. *Pattern Recognition*, 21(3):269–273, **1988**.

[111] Mohammed Nazim, Duong Trong, Sean Visal, and Jo Geun-Sik. Construction of semantic user profile for personalized web search. In *Proceedings of the 4th international conference on Computational Collective Intelligence: technologies and applications*, pages 99–108. **2012**.

[112] Stuart Middleton, Nigel Shadbolt, and David Roure. Ontological user profiling in recommender systems. *ACM Transactions on Information Systems*, 22(1):54–88, **2004**.

[113] Michael J. Pazzani and Daniel Billsus. Content-based recommendation systems, the adaptive web: methods and strategies of web personalization. In *Springer-Verlag*. **2007**.

[114] Xuehua Shen, Bin Tan, and Cheng Xiang Zhai. Implicit user modeling for personalized search. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management, CIKM 2005*, pages 824–831. **2005**.

[115] Song Wei, Zhang Yu, Liu Ting, and Li Sheng. Bridging topic modeling and personalized search. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1167–1175. ACM, **2010**.

[116] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. **1999**.

[117] David Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3(1):993–1022, **2003**.

[118] Nigam Kamal and Andrew Mccallum. Text classification from labeled and unlabeled documents using em. *The Journal of Machine Learning Research*, 39:103–134, **2000**.

[119] Eugene Volokh. Personalization and privacy. *Communications of the ACM*, 43(8):84–88, **2000**.

[120] Ayse Goker and T.L. McCluskey. Towards an adaptive information retrieval system. In *Proceedings of the 6th International Symposium, ISMIS'91*, pages 348–357. **1991**.

[121] Ingrid Zukerman and David Albrecht. Predictive statistical models for user modeling. *User Modeling and User-Adapted Interaction*, 11(1-2):5–18, **2001**.

[122] Bark Chiu, Geoffrey I. Webb, and Mark Kuzmycz. A comparison of first-order and zeroth-order induction for input-output agent modelling. In *Proceedings of the Sixth International Conference on User Modeling*, pages 347–358. **1997**.

[123] Brian Davison and Haym Hirsh. Predicting sequences of user actions. In *Notes of the AAAI/ICML 1998 Workshop on Predicting the Future: AI Approaches to Time-Series Analysis*. **1998**.

[124] Sofus Macskassy, Aynur A. Dayanik, and Haym Hirsh. Emailvalet: Learning user preferences for wireless email. In *Proceedings of the IJCAI99 Workshop on Learning about Users*, pages 45–50. **1999**.

[125] Joshua Alspector, Aleksander Koicz, and N. Karunanithi. Feature-based and clique-based user models for movie selection: A comparative study. *User Modeling and User-Adapted Interaction*, 7(4):279–304, **1997**.

[126] E. Gardner and Bernard Derrida. Optimal storage properties of neural network models. *Journal of Physics A: Mathematical and General*, 21(1):271–284, **1988**.

[127] Andrew Jennings and Hideyuki Higuchi. A user model neural network for a personal news service. *User Modeling and User-Adapted Interaction*, 3(1):1–25, **1993**.

[128] Judea Pearl and Stuart Russell. Bayesian networks. *In M.A. Arbib (Ed.), Handbook of Brain Theory and Neural Networks, Cambridge, MA: MIT Press*, pages 157–160, **2003**.

[129] Andre Berthold and Anthony Jameson. Interpreting symptoms of cognitive load in speech input. In *Proceedings of the Seventh International Conference, UM99*. **1999**.

[130] Piotr J. Gmytrasiewicz, Sanguk Noh, and Ted Kellogg. Bayesian update of recursive agent models. *User Modeling and User-Adapted Interaction*, 8(1):49–69, **1998**.

[131] Stephen C. Jameson and Jill Williams. Local needs and interventions for management of coral reefs in the developing tropical americas—the montego bay marine park case study. *Integrated coastal zone management of coral reefs : decision support modeling (English)*, pages 23–45, **2000**.

[132] Tessa Lau and Eric Horvitz. Patterns of search: analyzing and modeling web query refinement. In *Proceeding UM '99 Proceedings of the seventh international conference on User modeling*, pages 119–128. **1999**.

[133] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. Wiley; 1 edition (February 9, 1973), **1973**.

[134] Loc Nguyen. A user modeling system for adaptive learning. In *Proceeding of the 17th International Conference on Interactive Computer aided Learning (ICL2014)*, pages 864–866. **2014**.

[135] Rafael Morales Gamboa, Helen Pain, and T. Conlon. From behaviour to understandable presentation of learner models: a case study. In *Proceedings of the Workshop on Open, Interactive, and other Overt Approaches*. **1999**.

[136] Daniel Billsus and Michael J. Pazzani. A hybrid user model for news stories classification. In *Proceedings of the Seventh International Conference on User Modeling*, pages 99–108. **1999**.

[137] Chumki Basu, Haym Hirsh, and William Cohen. Recommendation as classification: Using social and content-based information in recommendation. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 714–720. **1998**.

[138] Diane Litman and Shimei Pan. Predicting and adapting to poor speech recognition in a spoken dialogue system. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pages 722–728. **2000**.

[139] Melinda T. Gervasio, Wayne Iba, and Pat Langley. Learning to predict user operations for adaptive scheduling. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 721–726. **1998**.

[140] Peter W. Foltz and Susan T. Dumais. Personalized information delivery: An analysis of information filtering methods. *Communications of the ACM*, 35(12):51–60, **1992**.

[141] Xuehua Shen, Bin Tan, and Cheng Xiang Zhai. Context-sensitive information retrieval using implicit feedback. In *Proceedings of SIGIR 2005*, pages 43–50. **2005**.

[142] Joachims Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. **1971**.

[143] Xuehua Shen and ChengXiang Zhai. Exploiting query history for document ranking in interactive information retrieval (poster). In *Proceedings of SIGIR 2003*, pages 377–378. **2003**.

[144] Amit Singhal. Modern information retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4):35–43, **2001**.

[145] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of 12th International World Wide Web Conference (WWW 2003)*, pages 271–279. **2003**.

[146] Ryen W. White, Joemon M. Jose, C. J. van Rijsbergen, and Ian Ruthven. A simulated study of implicit feedback models. In *Proceedings of ECIR 2004*, pages 311–326. **2004**.

[147] M.Rami Ghorab, Dong Zhou, Alexander O'Connor, and Vincent Wade. Personalised information retrieval: survey and classification. *User Modeling and User-Adapted Interaction*, **2012**.

[148] Nikolaos Nanas, Victoria Uren, and Anne De Roeck. Building and applying a concept hierarchy representation of a user profile. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, SIGIR '03*, pages 198–204. SIGIR, **2003**.

[149] Paul-Alexandru Chirita, Wolfgang Nejdl, Raluca Paiu, and Christian Kohlschütter. Using odp metadata to personalize search. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 178–185. ACM, **2005**.

[150] 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. ACM, **2012**.

[151] David Sontag, Kevyn Collins-Thompson, Paul N. Bennett, Ryen W. White, Sasuan Dumais, and Bodo Billerbeck. Probabilistic models for personalizing web search. In *Proceedings of the fifth ACM international conference on Web search and data mining - WSDM '12*, pages 433–442. ACM, **2012**.

[152] Milad Shokouhi, Ryen W. White, Paul Bennett, and Filip Radlinski. Fighting search engine amnesia: Reranking repeated results. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '13*, pages 273–282. SIGIR, **2013**.

[153] Liren Chen and Katia Sycara. Webmate: A personal agent for browsing and searching. In *Proceedings of the 2nd International Conference on Autonomous Agents*, pages 132–139. **1998**.

[154] Muhammad Zeb and Maria Fasli. Interest aware recommendations based on adaptive user profiling. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, pages 357–360. **2011**.

[155] Diane Kelly, Dollu Vijay Deepak, and Xin Fu. The loquacious user: A document independent source of terms for query expansion. In *Proceedings of the International Conference on Research and Development in Information Retrieval (SIGIR)*. **2005**.

[156] Eugene Agichtein, Eric Brill, and Susan Dumais. Improving web search ranking by incorporating user behavior information. In *Proceedings of the International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 19–26. **2006**.

[157] Yi Zhu, Li He, and Xiaojun Wang. User interest modeling and self-adaptive update using relevance feedback technology. In *2012 International Workshop on Information and Electronics Engineering*, pages 721–725. **2012**.

[158] Eduardo Vicente-López, Luis M. de Campos, Juan M. Fernández-Luna, and Juan F. Huete. Personalization of parliamentary document retrieval using different user profiles. In *Proceeding of the 2nd International Workshop on Personalization*. **2014**.

[159] Liu Yang, Qi Guo, Yang Song, Sha Meng, milad Shokouhi, Kieran McDonald, and Bruce Croft. Modeling user interests for zero-query ranking. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016*, pages 171–184. **2016**.

[160] Thanh Vu. *Dynamic User Profiling for Search Personalisation*. Ph.D. thesis, The Open University, **2017**.

[161] Hidekazu Sakagami and Tomonari Kamba. Learning personal preferences on online newspaper articles from user behaviors. *Computer Networks and ISDN Systems*, 29(8-13):1447–1455, **1997**.

[162] Henry Liebermann. Letizia: An agent that assists web browsing. In *Proceeding of the Intl. Conferences on AI*. **1995**.

[163] Henry Liebermann. Autonomous interface agents. In *Proceeding of ACM Conferences On Computers and Human Interaction (CHI'97)*. **1997**.

[164] Tomonari Kamba, Krishna Bharat, and Michael C. Albers. The krakatoa chronicle-an interactive, personalized newspaper on the web. In *Proceeding of 4th Intl. World Wide Web Conferences*, pages 159–170. **1995**.

[165] Dunja Mladenic. Personal webwatcher: Implementation and design. In *Technical Report IJS-DP-7472, Department of Intelligent Systems, J.Stefan Institute, Slovenia*. **1996**.

[166] Rob Barrett, Paul P. Maglio, and Daniel C. Kellem. How to personalize the web. In *Proceeding of ACM CHI'97*. **1997**.

[167] Ahu Siegg, Bamshad Mobasher, and Robin Burke. Learning ontology-based user profiles: A semantic approach to personalized web search. In *IEEE Intelligent Informatics Bulletin*. **2007**.

[168] James Allan. Incremental relevance feedback for information filtering. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 270–278. **1996**.

[169] Nawal Ould Amer, Philippe Mulhem, and Mathias Géry. Toward word embedding for personalized information retrieval. In *Neu-IR: The SIGIR 2016 Workshop on Neural Information Retrieval*. **2016**.

[170] Liana Razmerita, Albert Angehrn, and Alexander Maedche. Ontology based user modeling for knowledge management systems. In *Proceedings of the User Modeling Conference*, pages 213–217. **2003**.

[171] Thanh Tien Vu, Alistair Willis, Son Tran, and Dawei Song. Temporal latent topic user profiles for search personalisation. In *ECIR:37th European Conference on IR Research*, pages 605–616. ACM, **2015**.

[172] Thanh Vu, Dat Quoc Nguyen, Mark Johnson, Dawei Song, and Alistair Willis. Search personalization with embeddings. In *European Conference on Information Retrieval*, pages 598–604. Springer, **2017**.

[173] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc., **1999**.

[174] Minglai Shao and Liangxi Qin. Text similarity computing based on lda topic model and word co-occurrence. In *2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering (SEKEIE 2014)*. **2014**.

[175] Yang Song, Hongning Wang, and Xiaodong He. Adapting deep ranknet for personalized search. In *Proceedings of the 7th ACM international conference on Web search and data mining*. **2014**.

[176] Dai Nguyen, Thanh Vu, Tu Nguyen, Dat Nguyen, and Dinh Phung. A capsule network-based embedding model for search personalization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2180–2189. **2019**.

[177] Dai Nguyen, Dat Nguyen, Tu Nguyen, and Dinh Phung. A convolutional neural network-based model for knowledge base completion and its application to search personalization. *Semantic Web*, pages 1–14, **2018**.

[178] Saeedeh Momtazi and Fabian Lindenberg. Generating query suggestions by exploiting latent semantics in query logs. In *Journal of Information Science*, pages 437–448. ACM, **2016**.

[179] Das Abhinandan, Mayur Datar, and Ashutosh Garg. Google news personalization: Schable online collaborative filtering. In *WWW 2007/Track: Industrial Practice and Experience*, pages 8–12. **2007**.

[180] Thomas W. Malone, Kenneth R. Grant, Franklyn A. Turbak, Stephen A. Brobst, and Michael D. Cohen. Intelligent information sharing systems. *Communications of the ACM*, 30(5):390–402, **1987**.

[181] Pascal Chesnais, Matthew Mucklo, and Jonathan Sheena. The fishwrap personalized news system. In *Proceedings of IEEE 2nd International Workshop on Community Networking: Integrating Multimedia Services to the Home*, pages 275–282. **1995**.

[182] Yi-Shin Chen and Cyrus Shahabi. Automatically improving the accuracy of user profiles with genetic algorithm. In *Proceedings of IASTED International Conference on Artificial Intelligence and Soft Computing*, pages 283–288. **2001**.

[183] Joseph A. Konstan, Bradley N. Miller, David A. Maltz, Jon Herlocker, Lee R. Gordon, and John Riedl. Grouplens: Applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, **1997**.

[184] Beerud Dilip Sheth. *A Learning Approach to Personalized Information Filtering*. Ph.D. thesis, Massachusetts Institute of Technology, **1994**.

[185] Fabio Asnicar and Carlo Tasso. ifweb: a prototype of user model based intelligent agent for documentation filtering and navigation in the world wide web. In *Proceedings of the 6th Intl. Conf. On User Modeling*. **1997**.

[186] Mirco Speretta and Susan Gauch. Personalized search based on user search histories. In *Proceeding Of International Conference on Web Intelligence*, pages 622–628. **2005**.

[187] Kanoje Sumitkumar, Girase Sheetal, and Mukhopadhyay Debajyoti. User profiling trends, techniques and applications. *International Journal of advance foundation and research in computer(AJAFRC)*, 1(11):25–32, **2014**.

[188] Priyanka C. and Vinod Wadane Ghegade. A survey of personalized web search in current techniques. *International Journal of Computer Science and Information Technologies*, 5(6), **2014**.

[189] Alessandro Micarelli, Fabio Gasparetti, Filippo Sciarrone, , and Gauch Susan. Personalized search on the world wide web. *The Adaptive Web: Methods and Strategies of Web Personalization, Lecture Notes in Computer Science*, 4321, **2007**.

[190] Alexander Pretschner. *Ontology Based Personalized Search*. Ph.D. thesis, University of Kansas, **1999**.

[191] David M. Nichols. Implicit rating and filtering. In *Proceedings of the 5th DELOS Workshop on Filtering and Collaborative Filtering*, pages 31–36. **2006**.

[192] Jae-wook Ahn and Peter Brusilovsky. From user query to user model and back: Adaptive relevance-based visualization for information foraging. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 706–712. **2007**.

[193] Jae-wook Ahn and Peter Brusilovsky. Visual interaction for personalized information retrieval. In *Third Workshop on Human-Computer Interaction and Information Retrieval (HCIR 2009)*. **2009**.

[194] Jun Lai and Ben Soh. Personalized web search results with profile comparisons. In *Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05)*, pages 573–576. **2005**.

[195] Jun Lai and Ben Soh. Using element and document profile for information clustering. In *Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service*, pages 503–506. **2004**.

[196] Georges Dupret, Vanessa Murdock, and Benjamin Piwowarski. Web search evaluation using clickthrough data and a user model. In *Proceedings of the WWW Workshop on Query Log Analysis*. **2007**.

[197] Barry Smyth. A community-based approach to personalizing web search. *IEEE Journals, Computer*, 40:42–50, **2007**.

[198] Kenneth Wai-Ting Leung, Dik Lun Lee, and Wang-Chien Lee. Personalized web search with location preferences. In *IEEE 26th International Conference on Data Engineering*, pages 701–712. **2010**.

[199] Pasi Garbriella. Issues in personalizing information retrieval. *IEEE Intelligent Informatics Bulletin*, 11(1), **2010**.

[200] Jian-Tao Sun, Hua-Jun Zeng, Huan Liu, Yuchang Lu, and Zheng Chen. Cubesvd: A novel approach to personalized web search. In *Proceedings of the Fourteenth Intl. World Wide Web Conferences*. **2005**.

[201] Mehmet S. Aktas, Mehmet A. Nacar, and Filippo Menczer. Personalizing pagerank based on domain profiles. In *Proceedings of the WebKDD 2004: KDD Workshop on Web Mining and Web Usage Analysis*. **2004**.

[202] Jinxi Xu and Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. **1996**.

[203] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, **2002**.

[204] Zhongming Ma, Olivia R. Liu Sheng, and Gautam Pant. Evaluation of ontology-based user interests modeling. In *4th Workshop on e-Business*. **2005**.

[205] Bemard J. Jansen, Amanda Spink, Judy Bateman, and Tefko Saracevic. Real life information retrieval: A study of user queries on the web. *ACM SIGIR Forum*, 32(1):5–17, **1998**.

[206] Jay Budzik and Kristian Hammond. Watson: Anticipating and contextualizing information needs. In *Proceedings of the Sixty-second Annual Meeting of the American Society for Information Science*. **1999**.

[207] Eric J. Glover, Gary William Flake, Steve Lawrence, William Peter Birmingham, Andries Kruger, C. Lee Giles, and David M. Pennock. Improving category specific web search by learning query modifications. In *Proceedings 2001 Symposium on Applications and the Internet*. **2001**.

[208] Kyung-Joong Kim and Sung-Bae Cho. A personalized web search engine using fuzzy concept network with link structure. In *Joint 9th IFSA World Congress and 20th NAFIPS International Conference*. **2001**.

[209] Taher H. Havelivala. Topic-sensitive pagerank. In *Proceedings of the 12th International world Wide Web Conference (WWW 2003)*. **2003**.

[210] Zhongming Ma, Gautam Pant, and Olivia R. Liu Sheng. Interest-based personalized search. *ACM Transaction on Information Systems*, 25(5), **2007**.

[211] Paul-Alexandru Chirita, Claudiu Firan, and Wolfgang Nejdl. Summarizing local context to personalize global web search. In *Proceedings of the Annual International Conference on Information and Knowledge Management*, pages 287–296. **2006**.

[212] Paul-Alexandru Chirita, Claudiu Firan, and Wolfgang Nejdl. Personalised query expansion for the web. In *Proceedings of the ACM SIGIR Conference 2007*, pages 287–296. **2007**.

[213] Misha Bilenko, Ryen W. White, Matthew Richardson, and G. Craig Murray. Talking the talk vs. walking the walk: salience of information needs in querying vs. browsing. In *31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008)*, pages 705–706. **2008**.

[214] Sarabjot Anand, Patricia Kearney, and Mary Shapcott. Generating semantically enriched user profiles for web personalization. *ACM Transaction on Internet Technology*, 7(4), **2007**.

[215] Carsten Felden and Markus Linden. Ontology-based user profile. *Springer, Heidelberg (2007)*, 4439, **2007**.

[216] R. Guha, Rob McCool, and Eric Miller. Semantic search. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 20–24. **2003**.

[217] Mariam Ouanaim, Hamid Harroud, Aziz Berrado, and Mohammed Boulmalf. Dynamic user profiling approach for services discovery in mobile environments. In *Proceedings of the 6th IWC MC Conference*, pages 550–554. **2010**.

[218] Keith Bradley, Rachael Rafter, and Barry Smyth. Case-based user profiling for content personalization. In *Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 62–72. **2000**.

[219] Zhe Zhao, Zhiyuan Cheng, Lichan Hong, and Ed H. Chi. Improving user topic interest profiles by behavior factorization. In *1st International Workshop on Multilingual Web Access (MWA 2015)*. **2015**.

[220] Ryen W. White, Peter Bailey, and Liwei Chen. Predicting user interests from contextual information. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. **2009**.

[221] David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 32:61–70, **1992**.

[222] Ugur Çetintemel, Michael Franklin, and C. Lee Giles. Self- adaptive user profiles for large-scale data delivery. *ICDE*, **2000**.

[223] Stephen Robertson and Ian Soboroff. The trec-10 filtering track final report. *TREC-10*, **2001**.

[224] Dwi H. Widyantoro, Thomas R. Ioerger, and John Yen. An adaptive algorithm for learning changes in user interests. *CIKM*, **1999**.

[225] Tak W. Yan and Hector Garcia-Molina. Sift– a tool for wide- area information dissemination. *USENIX Technical Conference*, **1995**.

[226] Sviatoslav Braynov. Personalization and customization technologies. *Ch. 7, Wiley Publishing*, **2004**.

[227] K. Bollacker, Steve Lawrence, and C. Lee Giles. A system for automatic personalized tracking of scientific literature on the web. *ACM*, **1999**.

[228] Thorsten Joachims, Dayne Freitag, and Tom Mitchell. Webwatcher: A tour guide for the world wide web. *IJCAI*, **1997**.

[229] Michael J. Pazzani and Daniel Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, **1997**.

[230] Giovanna Castellano, Maria Fanelli, and Corrado Mencar. Similarity-based fuzzy clustering for user profiling. In *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops*, pages 75–78. **2007**.

[231] Rui Li, Wang Shengjie, Hongbo Deng, and Kevin Chen-Chuan Chang. Towards social user profiling: unified and discriminative influence model for inferring home locations. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1023–1031. **2012**.

[232] Pallavi Palleti, Harish Karnick, and Pabitra Mitra. Personalized web search using probabilistic query expansion. In *International Conferences on Web Intelligence and Intelligent Agent Technology Workshops (IEEE/WIC/ACM)*, pages 83–86. **2007**.

[233] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, **1958**.

[234] Chris D. Paice and Paul A. Jones. The identification of important concepts in highly structured technical papers. In *Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval*, pages 69–78. **1993**.

[235] Alberto Dıáz, Pablo Gervaś, and Antonio Garcıá. Evaluation of a system for personalized summarization of web contents. In *Proceedings of the tenth international conference on user modeling, LNAI 3538*, pages 453–462. **2005**.

[236] Anastasios Tombros and Mark Sanderson. Advantages of query-biased summaries in ir. In *Proceedings of the 21st ACM SIGIR conference*, pages 2–10. **1998**.

[237] Jaime G. Carbonell, Yiming Yang, Robert E. Frederking, Ralf D. Brown, Yibing Geng, and Danny Lee. Translingual information retrieval: A comparative evaluation. *IJCAI*, 1:708–715, **1997**.

[238] Jon Orwant. Heterogeneous learning in the doppelgänger user modeling system. *User Modelling and User-Adapted Interaction*, 4:107–130, **1995**.

[239] Mario Caruso, Massimo Mecella, and Roberto Baldoni. User profiling and micro-accounting for smart energy management. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 42. **2013**.

[240] John M. Carroll and Caroline Carrithers. Training wheels in a user interface. *Communications of the ACM*, 27(8):800–806, **1984**.

[241] Pat-Anthony Federico. Hypermedia environments and adaptive instruction. *Computers in Human Behavior*, 15:653–692, **1999**.

[242] J. R. Boyle Anderson and C. F. Reiser. Intelligent tutoring systems. *Science*, 228:456–468, **1985**.

[243] Cristina Conati, Abigail Gertner, and Kurt VanLehn. Using bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, 12:371–417, **2002**.

[244] Clare F. Harvey, Peter Smith, and Peter Lund. Providing a networked future for interpersonal information retrieval: Infovine and user modeling. *Interacting with Computers*, 10:195–212, **1998**.

[245] Susan Bull and Gord McCalla. Modelling cognitive style in a peer help network. *Instructional Science*, 30:497–528, **2002**.

[246] James Hendler. Is there an intelligent agent in your future? *Nature, Webmatters*, **1999**.

[247] Shari Trewin and Helen Pain. A model of keyboard configuration requirements. *Behaviour & Information Technology*, 18:27–35, **1999**.

[248] Bart Cornelis. *Personalizing search in digital libraries*. Ph.D. thesis, University of Maastricht, **Master's thesis CS 03-01**.

[249] Qi Zhiheng, Fan Nianbai, and Huang Zhenyu. User multi-interest modeling based on semantic similar network in personalized information retrieval. *Issues in Information Systems*, 10(2):257–266, **2009**.

[250] Tao Xiaohui, Li Yuefeng, and Zhong Ning. A personalized ontology model for web information gathering. *IEEE Trans. Knowledge Data Engineering*, 23(4):496–511, **2011**.

[251] Aniko Hannak, Sapiezynski Piotr, Arash Molavi Kakhki, Balachander Krishnamurthy, David Lazer, Alan Mislove, and Christo Wilson. Measuring personalization of web search. In *Proceedings of the 22Nd International Conference on World Wide Web, WWW '13*, pages 527–538. **2013**.

[252] Gerald Stermsek, Mark Strembeck, and Gustaf Neumann. A user profile derivation approach based on log- file analysis. In *International Conference on Information and Knowledge Engineering (IKE)*, pages 258–264. **2007**.

[253] Yoshinori Hijikata. Implicit user profiling for on demand relevance feedback. In *Proceedings of the 9th international conference on Intelligent user interfaces*, pages 198–205. **2004**.

[254] Christoph G. Thomas and Gerhard Fischer. Using agents to personalize the web. In *Proceedings of the 2nd international conference on Intelligent user interfaces*, pages 53–60. **1997**.

[255] Susan Dumais. Sigir 2003 workshop report: Implicit measure of user interests and preferences. *SIGIR Forum*, 37(2), **2003**.

[256] Nicolaas Matthijs and Filip Radlinski. Personalizing web search using long term browsing history. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 25–34. ACM, **2011**.

[257] Lance Spitzner. Honeypots. *Tracking Hackers. Addison Wesley Professional*, **2002**.

[258] Jiwei Li, Alan Ritter, and Eduard Hovy. Weakly supervised user profile extraction from twitter. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, **2014**.

[259] Ido Guy, Naama Zwerdling, Inbal Ronen, David Carmel, and Erel Uziel. Social media recommendation based on people and tags. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 194–201. SIGIR, **2010**.

[260] Ido Guy, Naama Zwerdling, David Carmel, Inbal Ronen, and Erel Uziel. Personalized recommendation of social software items based on social relations. In *Proceedings of the third ACM conference on Recommender systems*, pages 53–60. ACM, **2009**.

[261] Jingdong Wang, Zhe Zhao, Jiazhen Zhou, Hao Wang, Bin Cui, and Guojun Qi. Recommending flickr groups with social topic model. *Information retrieval*, 15(3-4):278–295, **2012**.

[262] Jilin Chen, Rowan Nairn, and Ed H. Chi. Speak little and well: recommending conversations in online social streams. In *Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011*, pages 217–226. SIGIR, **2011**.

[263] Pankaj Gupta, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Zadeh. Wtf: The who to follow service at twitter. In *Proceedings of the 22nd international conference on World Wide Web*, pages 505–514. **2013**.

[264] Kim Younghoon, Yoonjae Park, and Kyuseok Shim. Digtobi: a recommendation system for digg articles using probabilistic modeling. In *Proceedings of the 22nd international conference on World Wide Web*, pages 691–702. **2013**.

[265] Beidou Wang, Can Wang, and Jiajun Bu. Whom to mention: expand the diffusion of tweets by@ recommendation on micro-blogging systems. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1331–1340. **2013**.

[266] Erez Shmueli, Amit Kagian, Yehuda Koren, and Ronny Lempel. Care to comment?: recommendations for commenting on news stories. In *WWW 2012*, pages 429–438. **2012**.

[267] S. Rose, Dave Engel, Nick Cramer, and Wendy Cowley. Automatic keyword extraction from individual documents. *Text Mining: Applications and Theory*, pages 1–20, **2010**.

[268] Bougouin Adrien, Boudin Florian, and Daille Béatrice. Topicrank: Graph-based topic ranking for keyphrase extraction. In *Proceedings of the IJCNLP*, pages 543–551. **2013**.

[269] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into texts. In *Proceedings of the EMNLP 2004 (ed. Lin D and Wu D)*, pages 404–411. **2004**.

[270] Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig Manning. Kea: practical automatic keyphrase extraction. In *ACM conference on digital libraries*, pages 254–255. ACM, **1999**.

[271] Thuy Dung Nguyen and Minh-Thang Luong. Wingnus: Keyphrase extraction utilizing document logical structure. In *Proceedings of the SemEval*, pages 166–169. **2010**.

[272] Wan Xiaojun and Xiao Jianguo. Single document keyphrase extraction using neighborhood knowledge. *AAAI*, 8:855–860, **2008**.

[273] Matsuo Yutaka and Ishizuka Mitsuru. Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(1):157–169, **2004**.

[274] Hemant Misra, François Yvon, Olivier Cappé, and Joemon Jose. Text segmentation: A topic modeling perspective. *Information Processing & Management*, 47(4):528–544, **2011**.

[275] Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems*. **2009**.

[276] Tuomo Kakkonen, Niko Myller, Erkki Sutinen, and Jari Timonen. Comparison of dimension reduction methods for automated essay grading. *Natural Language Engineering*, 1:1–16, **2005**.

[277] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101:5228–5235, **2004**.

[278] Istvan Biro, Dávid Siklósi, Jacint Szabó, and András A. Benczúr. Linked latent dirichlet allocation in web spam filtering. In *Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web*. **2009**.

[279] Thanh Tien Vu, Alistair Willis, and Dawei Song. Modelling time-aware search tasks for search personalisation. In *Proceedings of the 24th International Conference on World Wide Web*, pages 131–132. ACM, **2015**.

[280] Thanh Vu, Alistair Willis, Udo Kruschwitz, and Dawei Song. Personalised query suggestion for intranet search with temporal user profiling. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, pages 265–268. ACM, **2017**.

[281] Christopher D. Manning, Prabhakar Raghavan, and Schütze Hinrich. Introduction to information retrieval. In *Introduction to Information Retrieval*. Cambridge University Press., **2008**.

[282] Samira Karimi Mansoub, Gönenç Ercan, and İlyas Çiçekli. Selective personalization and group profiles for improved web search personalization. *Turkish journal of electrical engineering & computer sciences*, 28(3):1631–1643, **2020**.