# PERSONAL ADVERTISEMENT RECOMMENDATION FOR MICROBLOGS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ATAKAN ŞIMŞEK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
COMPUTER ENGINEERING

JANUARY 2019

Approval of the thesis:

**PERSONAL ADVERTISEMENT RECOMMENDATION FOR MICROBLOGS**

submitted by **ATAKAN ŞIMŞEK** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Prof. Dr. Pınar Karagöz
Supervisor, **Computer Engineering, METU** _____

**Examining Committee Members:**

Prof. Dr. Erdoğan Doğdu
Computer Engineering, Cankaya University _____

Prof. Dr. Pınar Karagöz
Computer Engineering, METU _____

Prof. Dr. İsmail Hakkı Toroslu
Computer Engineering, METU _____

Prof. Dr. Ferda Nur Alpaslan
Computer Engineering, METU _____

Prof. Dr. Ahmet Coşar
Computer Engineering, THK University _____

Date: 24.01.2019

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name, Surname: Atakan Şimşek

Signature:

# ABSTRACT

## PERSONAL ADVERTISEMENT RECOMMENDATION FOR MICROBLOGS

Şimşek, Atakan
Ph.D., Department of Computer Engineering
Supervisor: Prof. Dr. Pınar Karagöz

January 2019, 110 pages

Advertisement recommendation on the Web is a popular research problem. For microblog platforms, different requirements arise due to the differences in the context of social media and social network. In this work, we propose an advertisement recommendation system for microblogs. The proposed solution uses all contents of the messages (texts, captions, web links, hashtags), and enhances them with sentiment data and followee/follower interactions expressed as microblog posts to generate a new user model. As another novel feature, Wikipedia Good Pages are used as general background knowledge for matching user profiles and advertisement contents. On the basis of the similarity between advertisement vectors and user profile vectors, the most related advertisement for the selected user is determined. Evaluation results show that the proposed solution performs better for advertisement recommendation on microblog platform and works faster in comparison to other techniques.

Keywords: Microblog, Advertisement, Recommendation, Wikipedia

# ÖZ

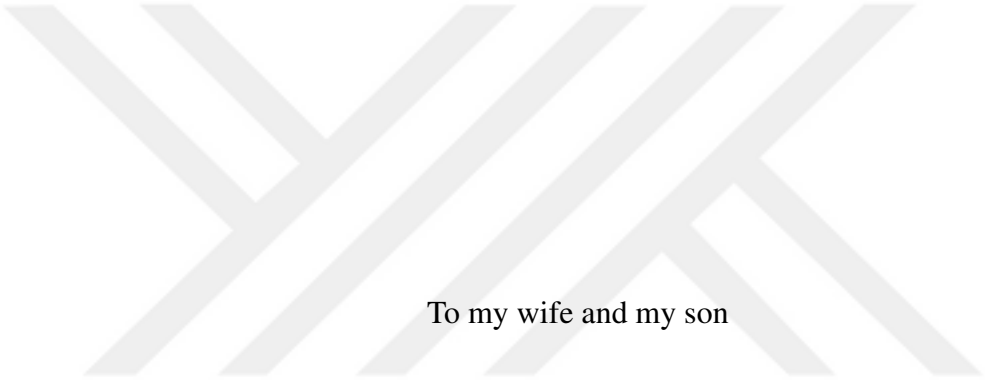
## MİKROBLOGLAR İÇİN KİŞİSEL REKLAM ÖNERİMİ

Şimşek, Atakan
Doktora, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Pınar Karagöz

Ocak 2019 , 110 sayfa

Internet ortamında reklam önerimi popüler bir araştıma problemidir. Sosyal medya ve sosyal ağlardaki farklılıklar var olan problemi mikrobloglar için başka ihtiyaçlara yöneltmiştir. Bu çalışmada, mikrobloglar için bir reklam önerim sistemini ileri sürmekteyiz. Önerilen çözüm mesajın bütün içeriklerini kullanmaktadır (metinler, etiketler, web sayfaları, iliştiriler), ve yeni bir kullanıcı modeli üretmek için duygu ve takipçi etkileşimini kullanmaktadır. Reklam içeriği ile kullanıcı profilindeki kelimeleri eşleştirmek için Wikipedia Good Pages ismi verilen sayfları kullanması bir diğer yenilikçi özelliğidir. Reklam kelime vektörü ve kişi profil vektörüne bakılarak ilgili kişiye en uygun reklama karar verilmektedir. Deneysel çalışmalar göstermiştir ki önerilen çözüm önceki çözümlerden daha iyi ve daha hızlı önerim yapmaktadır.

Anahtar Kelimeler: Mikroblog, Reklam, Önerim, Wikipedia



To my wife and my son

## ACKNOWLEDGMENTS

I would like to express my appreciation and indebtedness to my supervisor Prof. Dr. Pınar Karagöz for her supportive, encouraging and constructive approach throughout my Ph.D. study and her efforts during supervision of the thesis. I would like to thank to my thesis jury members Prof. Dr. Erdoğan Doğdu, Prof. Dr. İsmail Hakkı Toroslu, Prof. Dr. Ferda Nur Alpaslan and Prof. Dr. Ahmet Coşar for reviewing and evaluating my thesis.

I am also grateful to Ercan Güngör, Tuba Kesten, Selin Ünal, Tolga Avcı, Esen Tutaysalgır, A.Murat Özdemiray and Serap Şimşek for their valuable support during the evaluation phase of this thesis.

Finally, I would like to express my special thanks to my wife, my son and my family for their endless patience, encouragement and support during my Ph.D. study.

## TABLE OF CONTENTS

## LIST OF TABLES

TABLES

## LIST OF FIGURES

FIGURES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| IDF | Inverse Document Frequency |
| POS | Part of Speech |
| RS | Recommendation Systems |
| SNA | Social network analysis |
| SMM | Social Media Marketing |
| SVM | Support Vector Machine |
| TF | Term Frequency |
| TF-IDF | Term Frequency - Inverse Document Frequency |

# CHAPTER 1

# INTRODUCTION

## 1.1 Overview

Human beings discover that socializing in internet is easier than in real life. As a result of this, social networks have become very popular. Nowadays, nearly everyone especially in a particular age group has a social network profile. Microblog, which appeared at 2006, is one of the most popular social networking tools, right after blogging popularity. Compared to blogs, microblogging is the faster way of communication. Microblog takes its 'micro' name because of its character limit. The most popular microblog platform; Twitter has 280-character limit and 300+ million monthly active user count [11] at December 2018. The main difference between blog and microblog is ease-of-use. In microblogs, people can create content easily, there is no need to try investment for content generation. Another important difference is their update frequency. On average, a blog is updated in a couple of days, however a typical microblogger shares posts a couple of times in a day. In brief, microblogs allow users to broadcast short messages, sharing their statuses and opinions in the easiest way. Due to the character limit, users want to enrich their messages with hyperlinks, hashtags, videos and images. Another reason of its popularity is its accessibility. Microblogs can be used with mobile phones.

Beside providing users a platform to share and receive information, social network gives marketers a great opportunity to diffuse information/advertisements through a large population [5]. A significant ratio of marketers (%93) [12] indicate that they use social media as an advertisement diffusion tool. Social media allows marketers to make contact with the end-consumer at relatively low cost and higher levels of

efficiency [13]. Marketers always want to reach their customers in the fastest way. Internet is now the main message delivering medium between advertisers and consumers. It is a hot topic to find the best way of reaching the right customer in shortest path [14]. Broadcasting and fast communication are two properties of microblogs. As a result of this, marketers find out the power of microblogs and in recent years they effectively use microblogs. The name of this operation is Social Media Marketing (SMM).

Microblogs gain worldwide popularity recent years, this popularity makes microblogs as a potential of large information base. As a result of this, information extraction and knowledge discovery from microblogs become a hot topic in academic environments [15]. There are some traditional information extraction technologies and their successes are proved in Web corpus. Due to microblog specific structure and its distinct characteristics, classic methods cannot be applied easily. Therefore, information extraction from microblogs is a complex task and new algorithms can be developed. For example, in web documents and blogs, owner prepares a good text in order to explain some topic, they are well prepared documents. On the other hand, microblogs have some character count restrictions, moreover microblogger prepares a message at an instant time and most of times this messages are unstructured. It means there is no preparation time and there is no structure; moreover, sometimes abbreviations can be used instead of original words. i.e. 'ASAP' instead of 'As Soon As Possible', 'ps' instead of 'please see', 'BTW' instead of 'By The Way'. This usage, changes a normal message into noisy and ungrammatical text. It contains abbreviations, symbols and misspellings [15]. Classic NLP tools should be evolved and new algorithms should be developed in order to extract useful information from microblogs. Therefore, information extraction from microblogs is a hot topic and there are lots of new opportunities in this research area.

In other popular social networking sites; Facebook, MSN and MySpace, the communication graph is correlative. Two people should be 'friend' if they want to reach other person messages. On the other hand, in microblogs (most common microblog : Twitter) there is no such an obligation. Any person can follow any other person without any approval. It means there is a directed graph in microblogs unlike other common social network sites. Different graph representation means different graph

algorithms [15]. To summarize, microblogs have characteristic properties. Therefore, specific algorithms should be invented in order to use this large knowledge base in the most efficient way.

As we know, there is an unsolved problem about Social Media Marketing (SMM) recommendation for microblogs. Recent studies have some deficiencies and there is no complete solution about this problem. According to our studies, recommendation efficiency can be increased by using new approaches. Therefore, in this thesis, we aim to develop novel algorithms/techniques and compare the efficiency of these techniques with the previous studies in the field of advertisement recommendation to the microblog user according to the user interests. Differ from traditional recommendation techniques, we aim to focus and develop new algorithms on Wikipedia data usage and sentiment analysis usage in microblogs. As we know, Wikipedia data usage and sentiment analysis are the new topics in this domain, therefore there is only limited number of previous studies. We will develop novel representation models, which use all relevant microblog data, and they called as, user profile representation model and advertisement representation model. Consequently, users are not annoyed by irrelevant advertisements and relevant advertisements increase marketer click-through rates.

## 1.2 Background

In this section we will investigate the technologies and the studies which will be used as background information for this thesis.

### 1.2.1 Recommendation Systems

Recommendation systems are the subclasses of information filtering systems that recommend products, entertainment items or people which a user can be interested in [16]. Recommendation systems can be divided into two main sub-classes: Content Based Approaches and Collaborative Filtering Approaches. Content based approaches use a model which builds from properties of an item, on the other hand collaborative filtering approaches try to build a model from user social environment

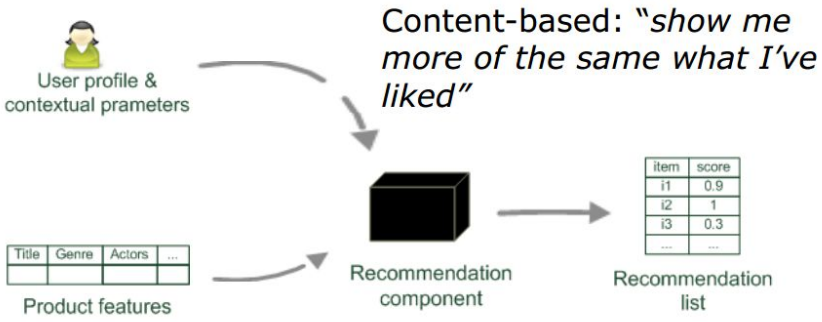



Figure 1.1: Content Based Recommendation [1]

[17].

#### 1.2.1.1 Content Based Approaches

This method uses the properties of items which will be recommended to the user [17]. Items are modeled according to their features. Items liked by the main user are analyzed. Candidate items are compared with the previously rated ones and the most similar items are recommended. In order to ensure similarity, items and user interests should be modeled in the same manner [18]. For model representation; according to user interests, item properties should be weighted. Direct feedback from a user can be used to assign different weights on the importance of certain attributes [17].

#### 1.2.1.2 Collaborative Filtering Approaches

In this kind of method, other users' choices are investigated to make a prediction pertaining to the selected user [19]. These algorithms assume that if people like same items in the past, they will like same items in the future. Among a group of people, some of them are selected according to interests whose interests are similar to the

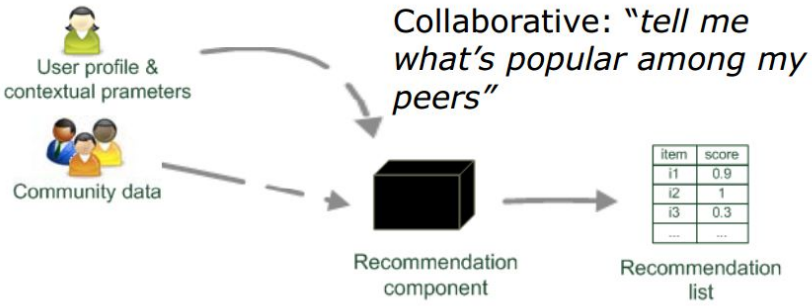



Figure 1.2: Collaborative Based Recommendation [1]

selected user and system uses their ratings to produce a prediction. This algorithm is different from content-based algorithms, because it does not know anything about the items, important things are people's opinions.

Main problems of these algorithms are 'first-rater' and 'cold-start'. First rater is a person who is new in the score/rating system, therefore the system does not know any preferences of this user. Cold start is a new item, new items never have been rated, therefore they cannot be recommended [19].

#### 1.2.1.3 Hybrid Recommendation Systems

There are some algorithm specific problems in content-based and collaborative filtering based approaches. Recent academic studies show that these problems can be resolved by combining these two main classes. Hybrid approaches can be implemented in several ways: algorithms can run separately and then their results can be combined. Content-based capabilities can be added to a collaborative-based approach (and vice versa); or the approaches can be unified into one model[17]. Recent studies show that hybrid methods provide better performance than pure content based methods and pure collaborative based methods.

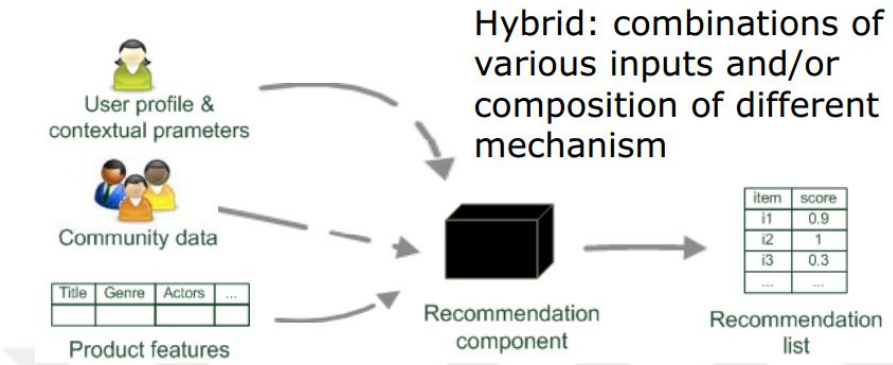



Figure 1.3: Hybrid Recommendation [1]

| | Pros | Cons |
|---|---|---|
| Collaborative | Nearly no ramp-up effort, serendipity of results, learns market segments | Requires some form of rating feedback, cold start for new users and new items |
| Content Content-based | Rating feedback easy to acquire, supports comparisons | Content-descriptions necessary, cold start for new users, no surprises |

Table 1.1: Recommender systems: basic techniques [1]

### 1.2.2 Microblogs and Twitter

In this section we will give a brief technical details about microblogs. There are several microblogs but one of them is a leading platform which is used in all countries of the world; Twitter. According to [20] Twitter global rank is 12 ("The rank is calculated using a combination of average daily visitors to twitter.com and pageviews on twitter.com over the past 3 months") at December 2018.

Therefore we will use Twitter as our development platform, because it is the most common microblog and there are only small differences between microblogs. i.e. Exact character limit, some of them allows to 160 char, some of them allows to 140 char. The main and most important difference between microblogs are followee-follower graph. Some of the microblogs only allows friend relationships. For example “Plurk” is a microblog and only two sided relationships are allowed. Fortunately, Twitter supports one sided relationship (anyone can follow whoever he wants) and two sided relationship is a subset of one sided relationship. Thus, our solutions are automatically applicable to this kind of microblogs.

People can broadcast a message, up-to 280 character, attach an image or a short video into his message. Besides creating a tweet, there are two other ways of communication. Comment a tweet or re-tweet. Any tweet can be commented by the follower of that person, or any message can be redistributing by re-tweeting this message. These operations empower a tweet diffusing ability, in order to reaching outside of one-degree subscribing network. Twitter supplies some abilities in order to extend twitter usage. ’@’ sign is used for initiating a directed conversation and ’#’ sign is used for easy and fast search operation. ’#’ and following text combination is called ’hash-tag’. It works as a link, when it is clicked, similar tweets which used same hashtags are listed [15]. Lastly, users can share a link with their followers, which means users can enrich their messages with anything they want. Moreover these links can be used for information extraction.

## 1.3 Motivation and Contribution

### 1.3.1 Problem Definition

Recommendation systems are very helpful for us, for example if you read a book and you like it, you want to get another book from the same author. Recommendation systems supply this property automatically and no physical effort is needed. Indeed, recommendation systems make your life easier. Microblogs are widely used social media tools, nearly everyone has an account and share some messages about their opinions. Therefore, the user explicit and implicit interests can be extracted from these platforms. If we have an intelligent system to extract user interests, we can easily recommend what the main user wants. As a result of this usage, 'Personal Advertisement Recommendation for Microblogs' is chosen as the main problem of our thesis.

Microblogs are new communication platforms and it has characteristics as described in Section 1.1, therefore traditional methods could not applied directly. New algorithms should be developed in order to extract useful information from microblogs. There are some academic studies about this issue as it can be seen in Chapter 2 but most of them is developed to increase marketers revenue. Mainly, they focus on Social Network Analysis to find correct people to spread marketers' products. They generate social graphs, investigate node characteristics, i.e. bridge node, close node, active node ..., hence they develop marketers' side of point algorithms. Some of them is developed for diffusing the specific advertisements in the fastest way, some of them is developed for scoring public opinions about their brand in microblogs.

On the other hand, we want to develop a system which recommends products for specific user in order to facilitate user life. It means, we want to develop a system for users. Our problem differentiates from former studies because we look on the reverse side. Nevertheless, our system will increase contextual marketing therefore marketers will gain also.

### 1.3.2 Motivation and Research Objective

With the rise of social media, companies want to exploit social media for their advertising. This article focuses on advertising on social media. The challenge it addresses is as follows: Given a set of advertisements, each of which is represented with a set of keywords, and a social media user, of whom the posts, followers and followees are known, recommend advertisements that the user is likely to click.

### 1.3.3 Contribution

The contribution of this study is twofold: The new user profile model is proposed and this user profile model is used in defining an advertisement recommendation method for microblogs. The proposed approach relies on several microblog features to generate an enhanced user model for profiling users' interests and the proposed system uses Wikipedia entries as a general background knowledge for matching the users' profiles and advertisement contents.

The main contributions of this thesis can be summarized as follows:

- The user profiling model is proposed to take into account microblogs artifacts (like captions, web links, and hashtags) as well as an influence model based on the followee/follower relationships and sentiment data. The model is completed and an article about this issue is published in [21]
- Using the profiling model, a recommendation algorithm is proposed for microblogs that:
  - is applicable to every set of advertisements
  - relies on Wikipedia Good Articles as points of reference
  - provides better recommendations and has better run time performances than previously proposed methods
- The recommendation algorithm is completed and published in [22]
- In order to enhance recommendation quality, the algorithm is enhanced with diversification.

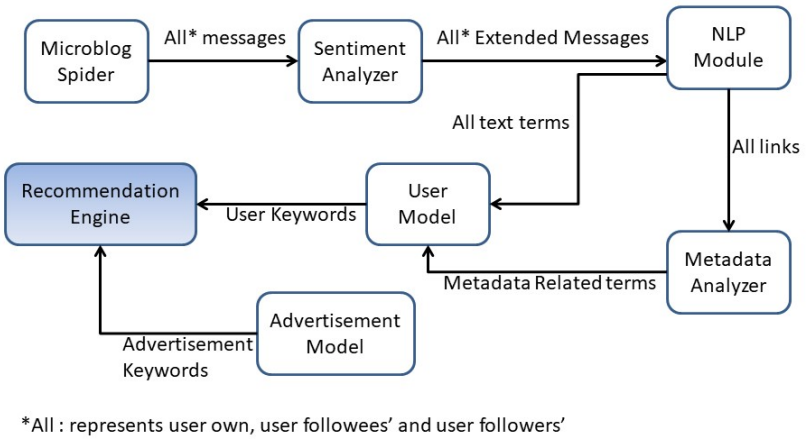



Figure 1.4: Our System Architecture

### 1.3.4 Proposed Work

In the previous section, we gave the definition of the problem. Figure 1.4 shows the road map of our solution. In this section, we explain the sub-problems and their proposed solutions:

- **Develop a novel user tweet representation model** by using message content, media (photo,video) data, hashtag, hyperlink information and meta data with the help of domain ontology and sentiment analysis. There are some studies about content based information extraction but these studies have some deficiencies. Some of them only concentrate on message contents, some of them only concentrate on hashtags, very few of them use combination of these contents. We develop a representation model which effectively uses the main user all data in order to recommend relevant advertisements. Moreover, we extract data from videos as a novel approach because there is no system which uses video data for information extraction in this domain. According to our studies, most of the video links are from YouTube and we pull data from YouTube page in order to enrich user profile data. These properties are our novel approaches at

content mining in microblogs. With the help of these new features, we propose a user profiling model which uses all valid data of the main user. Moreover, related works about this topic shows that sentiment analysis contribution is very poor in previous studies. We use sentiment data for user profiling to increase profiling capability.

- **Develop followee-follower influence model** which influence the main user. The solution of this sub-problem is similar with previous item. All followers and followees of the main user is analyzed, spammers and trolls are filtered. If a person has limited follower and lots of followee this type of accounts has no contribution to our study, therefore these accounts are filtered. The effect of the followers and followees are normalized according to interaction count.

- **Develop a disparate user preference model** which is a combination of user tweet representation model and followee-follower preferences. [7] shows that followee-follower tweets are more reliable than user own tweet to extract user hidden preferences. Based on this information, we develop a user preference model as a combination of user tweet representation model with followee-follower tweet representation model. [7] gives an equal weights into all followees and followers, but according to our survey this representation should be improved. We calculate weight of a friend by using interaction with this person and our specific user. Reciprocal messages, retweets, comments about their messages give a clue about influence coefficient. As a result, we develop a disparate user preference model which increase the efficiency of our system and results will be compared with the previous studies. Details of this model is explained in Chapter 3.

- **Develop advertisement representation model** in order to perfectly match user interests with advertisements. In former steps we develop a user preference representation model. To make an efficient recommendation, our advertisements should be modeled in similar ways. Some approaches use taxonomy tree for item and preference modeling. We want to improve this feature by using ontology data instead of taxonomy. As we know ontology contains more data than taxonomy, therefore ontology usage increases system efficiency. For example if a user express explicitly his favorite color is blue, we can concentrate on blue

items such as blue cars or blue clothes. In this thesis, an effective advertisement representation model is developed by using Wikipedia data. Details of this model is explained in Chapter 4.

### 1.3.5 Our Solution and Implementation

After literature survey, we decided the main architecture of our solution and we have completed our implementation. Efficient advertisement recommendation systems should have two important pieces; intelligent user interest model and intelligent advertisement model. If you want to make a good recommendation, these two models should perfectly match.

#### 1.3.5.1 Intelligent User Interest Model

User profile construction through keyword extraction from microblogs can be divided into two steps: preprocessing and keyword extraction. The type of preprocessing applied on the data is very effective on the quality of keyword extraction. For this reason, we firstly present the details of preprocessing step, and then describe the proposed semantic enhancement on hybrid TF-IDF. The details of this issue will be explained in Chapter 3 (User Profiling).

#### 1.3.5.2 Intelligent Advertisement Model and Recommendation

We proposed that, in microblog domain we increase the advertisement recommendation quality for individual users. To reach this goal; ontology data, metadata and user profile should be used in our advertisement model. As it can be easily inferred that, more advertisement data makes a better recommendation. Therefore, we want to increase advertisement knowledge. The details of this issue will be explained in Chapter 4 (Advertisement Recommendation).

## 1.4 Organization

The rest of this document is organized as follows. In Chapter 2, we will summarize some related work about our research topic. In Chapter 3, we will explain the user profiling part of the proposed work. In Chapter 4, we will describe the advertisement model and recommendation part of the thesis, in Chapter 5, we will enhance the recommendation quality by using diversification property. In Chapter 6, we will investigate the experiments & results, and in Chapter 7, we will conclude by summarizing our study.

# CHAPTER 2

# RELATED WORK

It this chapter, we will give a brief summary about some academic studies addressing information extraction from microblogs and recommendation studies in this domain.

## 2.1 Research on Content Mining at Microblogs

According to our survey, there are lots of studies about content mining at microblogs. Some of them collect data to recommend related news/related tweets, some of them use data mining to recommend followees. Unfortunately, there are limited number of studies, which use content mining to recommend advertisements. Therefore, we investigated other domains' (news/tweet recommendation) academic papers and summarized these papers in this section. According to these papers, content mining process can be divided into two parts. Preprocess and information extraction steps. Details of these steps can be varied according to academic study quality, therefore we will investigate about general approaches. In the first step, tweets are processed; reply messages are pruned. If academic study doesn't make sentiment analysis; emoticons, stopwords, links and media items are removed. Some studies use part of speech tagger to tag words. To sum up, many of these academic studies have a very limited preprocess step. Therefore, lack of above operations decreases the recommendation quality. Moreover according to our research, link and media item usage is very rare in microblogs. Lastly, there is no academic study which use location data to recommend some products.

In the second step, information is extracted by different algorithms. Most popular algorithm is TF-IDF ranking. TextRank, PageRank, Ngram with tag, LexRank, Mead,

Cluster, MostRecent and SumBasic are the other algorithms in this problem domain. Some research papers compare these algorithms in their test set. We can conclude that Hybrid TF-IDF and Pagerank are the most valuable algorithms in order to solve information extraction problem solution. In our implementation part we will investigate these algorithms' details. Some studies about this topic:

Yung-Ming Li *et al.* [5] proposed a diffusion mechanism to deliver advertisement information over microblogging. The design of the diffusion mechanism is conceptually similar to the computer network multicast methods. Multicast is a network technology for the delivery of information to a specific group with the most efficient strategy.

For an advertisement, provided by a sponsor/marketer, AdPlurker will select the top-k most suitable users as the endorsers. These endorsers are invited from the Plurk users. In order to join this experiment, user should first follow the AdPlurker account in Twitter. After selecting the first set of users, AdPlurker sends private messages, which contain the advertisement, to these users. Selected users receive a message and a recommended list of users who are potentially interested in the advertisements. When a social endorser sends the advertisements to their friends which are received from AdPlurker, the system triggers the endorser discovery. This engine sends a recommended list of friends to each endorser. This cycle continues until it reaches the desired number of people.

Their work differs from existing works in some aspects. In this work; rather than traditional web sites, they use microblogs as the data source. Moreover, they apply recommendation mechanism to online social advertising in order to diffuse a selected advertisement in the fastest way [5]. To find the right endorsers, they use four properties.

1. User Preference Extraction: The preference tree of an individual user is constructed based on a predefined category tree. User preferences can be collected by online questionnaires or fan page of a particular brand. It means there is no automatic system to construct user preference tree. This part is evaluated manually from questionnaires.

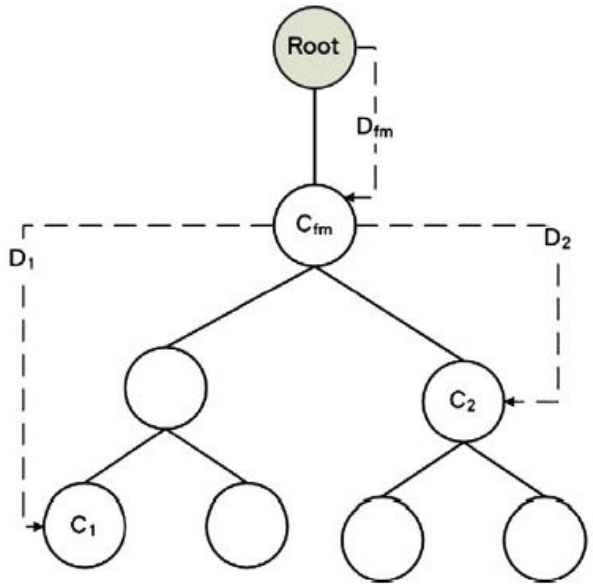



Figure 2.1: Category Tree [5]

2. Advertisement Fitness: They establish the category tree of advertisements and use the same tree to define user preferences. They use distance-based approaches to find similarity between user preferences and advertisements.

   The preference score is measured based on the distance of the shortest path between the preference category of a user and the product category of an advertisement. If an advertisement belongs to two or more categories, the preference score will be the average value. $C_1$ and $C_2$ stand for the target category of the advertisement. $C_{fm}$ represents the first mutual parent node of $C_1$ and $C_2$ in a catalog tree. The fitness degree of the advertisements to a user can be calculated by the equation 21. In this equation, $D_1(D_2)$ is the length of the path from $C_1(C_2)$ to $C_{fm}$ and $D_{fm}$ is the distance of the path from $C_{fm}$ to the root node in the category tree [5].

$$Sim_p(C_1.C_2) = \frac{2D_{fm}}{D_1 + D_2 + 2D_{fm}} \tag{21}$$

3. Influence Analysis: This is an advertisement diffusion framework, therefore Social Network Analysis (SNA) is used. Users and their social network graphs are analyzed and most suitable people are selected to diffuse advertisement. For

example, it is found that the out-degree centrality is better on message diffusion.

4. Propagation Strength: User social activity and social interactions are investigated. For example, activity of a user is calculated by the number of posts during a period of time in the social platform.

Using the four criteria, which are explained above, they suggest a user list to diffuse a specific advertisement. It is a marketer side academic study, to inform more users in the fastest way about an advertisement. Evaluation results show that, there is an increase in exposure and resonance values.

I-Hsien Ting *et all.* [6] proposed a framework to analyze the opinion groups in blogosphere. This methodology has two phases. In the first phase, the model is trained. In the second phase, opinion groups are identified. Training data is selected as the pre-defined data from blogs for a special event. In this study, there are only two classification groups, positive and negative. The training data is selected by experts, in order to model positive opinion and negative opinion. Then, data will be analyzed using content mining techniques and keywords are extracted according to Term Frequency - Inverse Document Frequency (TF-IDF). This TF-IDF values are used to calculate the rank of keywords (Figure 2.2). Moreover, users - group relationship matrix is calculated by counting the the responses to an article in the blogs.

For SNA (Social Network Analysis), they use four metrics; degree centrality, closeness centrality, betweenness centrality and cluster coefficient. Five topmost keyword and four SNA metrics define the model as seen in the Figure 2.3.

The similarity of the each user and a model will be measured by using the cosine similarity. When the similarity is higher than a threshold, it will be classified as positive or negative opinion about this special event.

Chen Xu *et al.* [7] proposed a solution for detecting user preferences on Microblogs. They claim that users might be reluctant to express their feelings and preferences on their messages due to privacy issues. On the other hand, user interests are very important for business values, i.e. advertisement recommendation. They propose that user inner feelings (preferences) never be concealed in those information they read, especially in microblogs. For that reason, they use followees' messages besides the user's

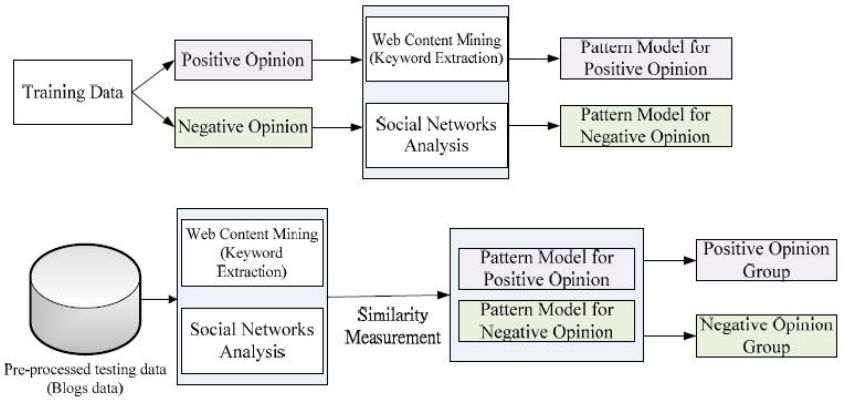



Figure 2.2: Opinion groups research methodology [6]

| | **Positive pattern model** | **Negative pattern model** |
|---|---|---|
| **Keyword1 (TF-IDF)** | Great (0.512) | Disagree (0.677) |
| **Keyword2 (TF-IDF)** | Good decision (0.341) | Unreasonable (0.336) |
| **Keyword3 (TF-IDF)** | Nice (0.107) | Stop (0.111) |
| **Keyword4 (TF-IDF)** | Understand (0.044) | Stupid (0.097) |
| **Keyword5 (TF-IDF)** | Like (0.035) | Ridiculous (0.055) |
| **Degree centrality** | 2.414 | 4.112 |
| **Closeness centrality** | 1.741 | 2.121 |
| **Betweeness centrality** | 2.322 | 3.819 |
| **Cluster coefficient** | 2.865 | 4.7 |

Figure 2.3: The pattern models of positive and negative opinion groups [6]

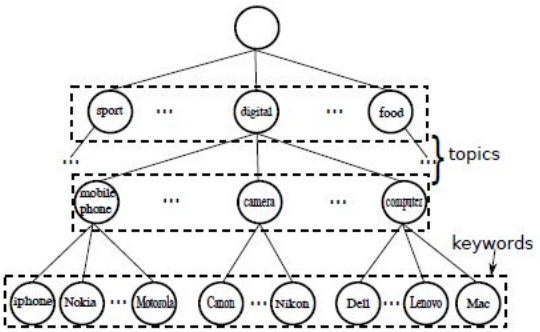



**Table 1.** Preference Vector

| *topic* | sport, $\cdots$ digital, $\cdots$ food | *total* |
|---|---|---|
| *vector* | ( 0.2, $\cdots$ 0.36, $\cdots$ 0.01 ) | 1 |

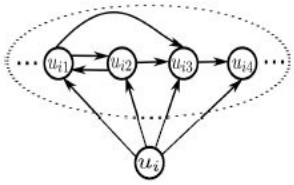



Figure 2.4: Tree structured taxonomy and preference vector [7]

own messages in their user preference model. Some kinds of microblogs(Twitter, Sina) use unilateral relationship instead of bilateral ones (Facebook). This property leads to an opinion; reading the messages from his followees' pages, reflects user internal feelings about this issue. In this type of microblogs, user can follow anyone because getting permission from the opposite side is not an obligation. Therefore user followee lists represent user's real preferences. They use Chinese language specific taxonomy with the leaf nodes as keywords and non-leaf nodes as topic (Figure 2.4).

They([7]) claim that followee tweets are more important than a user's own tweets to extract user preferences. Therefore they develop a tweet signature model and extract followees' tweet signature. In this equation $u_{ij}$ is one of the $u_i$'s followee and $u_{ij}$ tweet signature is defined as Eqn 22 where $\theta_{n,u_{i,j}}$ is $u_{ij}$ preference level on the item $\theta_n$. As seen in the Figure 2.4 items in this vector are the set of top level items in the taxonomy. Moreover, the preference level $\theta_{n,u_{ij}}$ is the sum of weighted term frequency of all descendants of $\theta_n$

$$\Theta_{u_{ij}} = (\theta_{1,u_{ij}}, \theta_{2,u_{ij}}, ..., \theta_{n,u_{ij}}) \tag{22}$$

Lastly they define user preference as in Equation 23, in this equation $\beta$ represents the experiment coefficient. $\beta$ = 0 means only user's own tweets are used, $\beta$ = 1 means only followees' tweets are used to extract user preference vector. $Inf(u_i, u_{ij})$ means influence over $u_i$. In this experiment, $Inf(u_i, u_{ij})$ = $\frac{1}{Numberoffollowee}$. Their results show that the case with $\beta$ = 1 gets the highest score and the case with $\beta$ = 0 gets the lowest score. Consequently, the tweets from followees are more important than the user's own tweets to extract user preferences.

$$\mathbf{P}_{u_i} = (1 - \beta)\Theta_{u_i} + \beta \sum Inf(u_i, u_{ij})\Theta_{u_{ij}} \tag{23}$$

Weishi Zeng *et al.* [23] proposed a framework for information collection and information release. Their system is developed in order to magnify marketing effects and improve marketing strategies. Their work mainly focus on Social Network Analysis and graph theories. Like previous studies, this study is a marketer side framework, therefore individual users are not important. Groups of people and their characteristics are main issue in their paper. Their work can be divided into two parts; information collection on content level and information collection on network level. In the first part, microblog search functionality is used for learning users opinion about brands and companies. Companies can develop marketing strategies manually by investigating this search result. In the second part, SNA is done and some values are collected; Clustering Coefficient, Centrality, Bridge Centrality and Clique. For SNA, all these terms have an definition and these values give good clues about user's roles in the social network structure. In this study, with using these roles and positions new marketing strategies are proposed.

Delip Rao *et al.* [2] proposed a novel investigation of binary classification algorithm for extracting latent user attributes in Twitter. They develop this framework to extract user attributes, like age and gender, that are directly important for providing personalized services. They conclude that, the status message(tweet) contents are more

| Feature | female / male |
|---|---|
| Emoticons | 3.5 |
| Elipses | 1.5 |
| Character repetition | 1.4 |
| Repeated exclamation | 2.0 |
| Puzzled punctutation | 1.8 |
| OMG | 4.0 |

Table 2.1: Sociolinguistic-based features for Gender expressed as relative frequency of females and males [2]

valuable than social network structure. Therefore, they mainly focus on message contents.

The differences in lexical choice and other linguistics features expose user latent attributes such as; sequence of exclamation marks usage is more common for female users than male users. People laugh differently on Twitter, women use LOL, men use LMFAO. 'Dude' and 'bro' words distinguish younger users from older users. They found that, female users employ character repetition like 'niceeee' 1.4 times more than male users, detailed results can be seen in Table 2.1.

I-Hing Ting *et al.* [24] proposed a solution to understand user preference to assist marketers for target marketing. User's messages content and social structure is used in this approach. They develops their system by using Plurk which is a common microblog in Thailand. Plurk differs from other microblogs because it supports two way edges which means only friend relationship is allowed like Facebook. Therefore, Plurk social graph is a subset of Twitter graph.

In SNA, most common measurements are; density, closeness, centrality, and betweenness. SNA values can be used to define social role of the selected user. For example 'social user' can be defined by higher out-degree. On the contrary, a 'star' can be defined by higher in-degree value. In the study of [24], they combine content mining data with SNA data in order to enhance recommendation quality. They use this system to recommend marketers whose products can be interested by these users. To put

it another way, they suggest to recommend list of the target users for companies. If a company wants to diffuse a product, this system will find the most suitable target users to extend marketers' products.

Their system has two main modules; content analyze module and social network analyze module. In the content analyze module, system collects the users messages and eliminate the stop-words. They use CKIP (Chinese Knowledge Information Processing) a morphological analysis tool to define part of the speech. Keywords are extracted by using TF-IDF process and extracted keywords are stored into the database. They sorts keywords according to their TF-IDF values and top-k keywords are selected for modeling user preferences.

They use some formulas to model users, categories and products but this framework is not implemented yet therefore there is no evaluation and no real data.

Mihalca and Tarau [25] proposed an algorithm called TextRank for processing texts. TextRank is a graph based, unsupervised algorithm for keyword and sentence extraction. According to authors, in graph based ranking models, basic idea is voting or recommendation. Words are simulated by vertices and their co-occurrences are simulated by edges. If a vertex is linked with another vertex, it means these vertices vote each other. Higher number of votes means higher number of importance. In the formula 24, G = (V,E) is a directed graph, with vertices V and edges E, E is the subset of the $V * V$. For a vertex $V_i$, $In(V_i)$ represents the incoming edges vertices. $Out(V_i)$ represents outgoing edges vertices. d is the damping factor that can be selected between 0 and 1,usually set to 0.85. Score of the vertex $V_i$ is calculated in the formula 24 [25].

$$S(V_i) = (1-d) + d * \sum_{j \in (In(V_i))} \frac{1}{Out(V_j)} S(V_j) \tag{24}$$

If we analyze the TextRank score formula, we realize that value is calculated iteratively. For each iteration, values are changed and after enough iteration, value is converged. Initial value selection is the main problem of this formula. Fortunately, according to authors, final score is not affected by initial values. To enhance solution, this formula can be used for other types of graphs. For example, for undirected graph,

out degree of a vertex is equal to the in degree of the vertex. For weighted graphs, formula can be changed as 25

$$WS(V_i) = (1-d) + d * \sum_{V_j \in (In(V_i))} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \tag{25}$$

For text summarization or keyword extraction, natural language texts can be defined as graphs. Keywords are vertices and neighborhood can represents edges. In their paper, they use co-occurrence relation between keywords, by controlling the distance between words. They define that, two vertices are connected if these vertices co-occur between 10 words window. After textrank score calculation, they summarize the texts and extract keywords.

Wu *et al.* [3] proposed a solution to automatically extracting annotations from user micro-blog. They applied TFIDF ranking algorithm and TextRank algorithm to tagging user tweets and user interests. They thought that extracting user interest can be very helpful for commercial products. For example, opinions about a company's products can be used as feedback. They formulate this problem as a keyword extracting task, by selecting correct keywords from user tweets. They made preprocess operations and they used TFIDF and TextRank algorithm. According to their result, system gives good results with TextRank algorithm on randomly selected users.

Their frameworks can be seen in Figure 2.5. Their preprocessing part can be divided into 5 different part. 1) They remove reply tweets from corpus, because they think reply tweets contains more information about replied person. 2) They remove emoticons. In this work, sentiment analysis is not used, therefore emoticons are irrelevant. 3) Twitter is a noisy platform, there are lots of slang words and abbreviations. In this study, they create a look-up table and they change these words with grammatical regular words. 4) They use Stanford POS tagger, because only nouns and adjectives are used for keyword extraction process. 5) Lastly, words are processed with Porter stemmer and stop-words are removed at the end of the preprocessing step.

For experimental setup, they used Twitter as microblog and for an interested topic they searched the Twitter, randomly 11.376 user was selected and each had 180-200

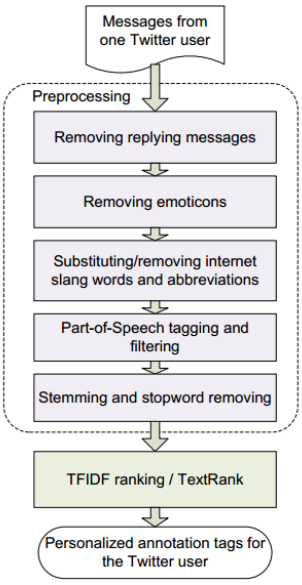



Figure 2.5: Personalized annotation tag framework [3]

| Precision | TFIDF | TextRank |
|---|---|---|
| top-1 | 59.6 | 67.3 |
| top-3 | 61.5 | 66.0 |
| top-5 | 61.2 | 63.0 |
| top-10 | 59.0 | 58.3 |

Table 2.2: Tagging precision on users [3]

tweets. They used two different user tag extraction methods; TF-IDF ranking and TextRank algorithms. In the TF-IDF ranking algorithm all tweets of a person called as a document and sum of all people tweets form the document collection. In the TextRank algorithm, they used weighted undirected TextRank method. They added an edge if two keyword occurs in a tweet and weight of this edge is counted as total count of within message.

They compared top-N precision of these algorithms and they had human judges to extract the output tags of selected users. Experimental results were summarized in the Table 2.2.

Inouye and Kalita made a survey about Twitter summarization algorithms and they

| | F-measure | Recall | Precision |
|---|---|---|---|
| LexRank | 0.2027 | 0.1894 | 0.2333 |
| Random | 0.2071 | 0.2283 | 0.1967 |
| Mead | 0.2204 | 0.3050 | 0.1771 |
| Manual | 0.2252 | 0.2320 | 0.2320 |
| Cluster | 0.2310 | 0.2554 | 0.2180 |
| TextRank | 0.2328 | 0.3053 | 0.1954 |
| MostRecent | 0.2329 | 0.2463 | 0.2253 |
| Hybrid TF-IDF | 0.2524 | 0.2666 | 0.2499 |
| SumBasic | 0.2544 | 0.3274 | 0.2127 |

Figure 2.6: Average values of F-Measure, Recall and Precision ordered by F-Measure [8]

improved TFIDF algorithm performance by adding hybrid property into this algorithms [8]. Their new algorithm made same calculations with original TFIDF algorithm but they change document and document collection definition in this algorithm. According to their solution, they define a document as a single post, but when calculating word frequency they assume that a document is the entire collection. This solution increases TF-IDF quality, moreover they compare keyword extracting algorithms and Hybrid TF-IDF is the one of the top performance algorithms(Figure 2.6).

## 2.2 Research on Sentiment Analysis and Domain Ontology

Ontology usage is obviously increased the recommendation quality, therefore there are lots of study in this topic. However, advertisement recommendation in microblogs rarely use ontology data, because former studies mainly focused on recommending the specific products. As we said earlier, previous studies look this problem at the marketer side. They have some specific products and they want to diffuse these products to many people as soon as possible. On the other hand, we want to use domain ontology data in order to recommend the true product. For example if a person buys an Ipad and he/she is very happy with this product, we should recommend a cover in order to protect his/her property. If a person buys a new laptop, we can recommend a mouse to this person.

Sentiment analysis is an important issue which increases recommendation quality. Previous studies use sentiment analysis for finding users/public opinions about selected products. They gather statistical data about users profiles and find positivity

or negativity about selected products. According to our studies, there is no such a system which analyzes user tweets and recommend some products according to this sentiment data. In order to use this deficiency we use sentiment analysis at information extraction operation. Twitter has two types of sentiment data, emoticons and word sentiments. Emoticons are some symbols which express the user feelings, i.e :) . Word sentiments are adjective words which are used in sentences to express user moods i.e. very good. Some studies about this topic:

Eric Cambria *et al.* [26] proposed a framework to analyze sentiment data for Social Media Marketing. They claim that previous studies rely on the part of text in which emotional states are explicitly expressed, therefore previous studies cannot capture implicit opinions and sentiments that are expressed explicitly.

Their framework can be divided into three modules; NLP module, Semantic Parser and AffectiveSpace module. NLP module searches the entire text to find emotional items like; special punctuation, complete upper-case words, onomatopoeic repetitions, exclamation words, negations, degree adverbs and emoticons. Semantic Parser module deconstructs text into concepts. AffectiveSpace module calculates the sentic vector of each concepts according to Hourglass model.

Hourglass model is a variant of Plutchik's emotion categorization according to four different dimensions [26].

- the user is happy with the service provided (Pleasantness)
- the user is interested in the information supplied (Attention)
- the user is comfortable with the interface (Sensitivity)
- the user is disposed to use the application (Aptitude)

At Hourglass model, each dimension is characterized by six levels to define the intensity of the emotion grade. Moreover, to get better results they develop a human emotion ontology.

They evaluate their system by using LiveJournal [27], this website is a virtual community of more than 23 millions of users who keep a blog, journal and diary [26]. In this

website, bloggers are allowed to label their posts with a mood tag. For that reason, Eric Cambria *et al.* used this website as evaluation dataset. These blogs are classified by their system and results are compared with author labeled moods. According to their results, system has %70-%85 precision.

Yung-Ming Li *et al.* [28] develops a summarization framework that provides numeric summarization for microblog opinions. It is a marketer side framework, main goal of the project is extracting the quantitative opinion score (positive and negative) about a topic. Therefore marketers can investigate public opinion about their brands and their products. Their work can be divided into four steps:

1. Topic Detection: This step starts with a web spider which collects opinions from web about relevant topics around a user query. After opinions are collected, a POS-tagger (Stanford Part-of-Speech) is used. This module is designed to rank terms to find the most relevant terms. TFF-IDF property is used and the most relevant k-terms are extracted.

2. Sentiment Classification: In previous studies, sentiment classification is very weak, only two sets of clusters are done; positive and negative. In their work, they improve the previous works as giving a score to an opinion. This score represents opinion quality. They use Word-Net to prepare emotional and sentimental words. In order to convert text opinion to numeric values, they use Support Vector Machine (SVM). SVM is a supervised machine learning method, therefore a training set should be prepared. They use a very simple method to prepare the training set. In microblogs, if there is an emotional symbol ':)' it means positive set, if text contains ':(' it means negative opinion. If text contains both of them this opinion is ignored.

3. Credibility Assessment: This step is a filtering step, in this step 'trollers' and 'spammers' are ignored. To measure the credibility score they use follower-followee ratio.

4. Score Aggregation: At the last step, a weighted additive aggregation formula is used for aggregating score for a topic about a query.

Garcia Esparza *et al.* [29] proposed a system for a microblog whose name is Blipper.

Blipper is a service for reviewing the products from five different categories; applications, music, movies, books and games. Moreover, there are four types of emotion measurements; love it, like it, dislike it and hate it. It supplies 160-char text messages and tags for a product. To recommend products, this system extracts two indexes; Product index and user index. Product index is extracted by using TF-IDF values of messages' and tags' keywords. User index is extracted only using messages' keywords. They use Lucene index in order to extract most valuable terms.

Products terms and user terms are matched according to similarity measurement and recommendation is done by using this similarity. They use two different recommendation techniques; content based recommendation by using similar terms and collaborative based recommendation by using similar users preferences. According to their results content-based approaches give better results [29].

Meador and Gluck [4] proposed a solution for determining public opinion about interested topic by using microblogs. They mainly focused on movies and stocks. At the beginning, they used Twitter search API, but they realized that this API supplies only last 1500 tweet about search topic, because Twitter has some query limits. After that, they connected with the Twitter with real time(stream) API and they solved this problem by collecting tweets dynamically. At result, they created their test corpus with 2.5 million movie related tweets and 7.9 million stock related tweets. This corpus consists from eight different companies and four different movies.

In this study, they want to analyze public opinions, therefore they use subjectivity lexicon for finding sentiment words. Subjectivity lexicon is a list of 8200 subjective words for English language. Moreover, they use emoticons (Table 2.3) in order to enhance sentiment analyze, because emoticons are widely used in previous works. Emoticons are unambiguous and they are one of the most reliable signs for mood analysis.

In order to enhance sentiment level, they referenced another list; Twitrattr which is an online service to determine sentiment in the tweets. It has 150 subjective clues, but it is totally specialized to twitter jargon. It has better performance for noisy tweets(words) like LOL, FTL, pwnd... Moreover they add "not" keyword into Subjectivity Lexicon to cover negative values of adjectives. After these operations they

| | Emoticon |
|---|---|
| Positive | ;) :) :D =) |
| Negative | :( :”( :C =( |

Table 2.3: Emoticons [4]

calculate subjectivity score of selected tweet as Formula 26. In this formula they take the frequency of each subjectivity word $c_n$ from 1 to n and multiply this value by polarity(+1 or -1). A tweet score is represented by summing all these words scores.

$$s = \sum_{1}^{n}(freq(c_n) * polarity(c_n)) \tag{26}$$

They use two different calculation methods for interested item daily sentiment score; Formula 27 and Formula 28. According to their results, there is a significant relation between tweet count and box value of this movie(Figure 2.7).

$$f_d = \frac{\sum_{1}^{n}(positive_n)}{\sum_{1}^{n}(negative_n)} \tag{27}$$

$$f_d = \frac{count(positive)}{count(negative)} \tag{28}$$

Gabrilovich and Markovitch proposed a system which finds the similarity of a text and Wikipedia concepts [30]. Wikipedia concepts and any text are modeled as weighted list of words vector, therefore they use vector similarity. By that way, they find the most similar top N Wikipedia concepts. As a result, they can summarize any text by using these concept keywords.

Lu and Lam proposed a system which recommends tweets according to user interests [9]. Their system analyzes user previous tweets and extends tweet knowledge by Wikipedia data. They use random walk method to increase recommendation quality.

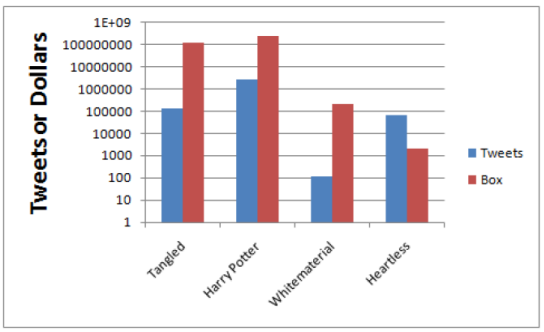



Figure 2.7: Distribution of tweets and box by movie [4]

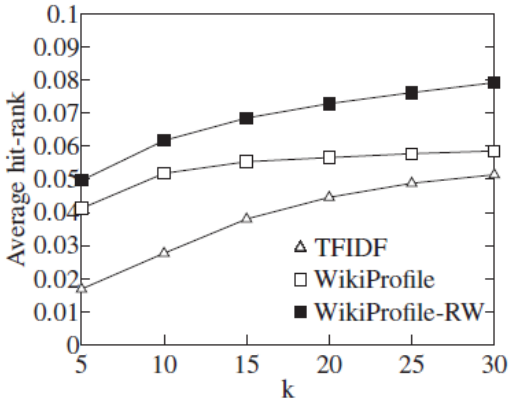



Figure 2.8: Recommendation performance measured by average hit-rank [9]

Moreover, their user model includes user friends to get better results. On the other hand, their model gives equal importance to user all friends, they ignore interaction frequency. After extracting user interests, a massive search is made in Twitter and best K matched tweets are recommended. Their evaluation graph can be seen in Figure 2.8, Wikiprofile and Wikiprofile-Random Walk increase the recommendation performance.

## 2.3 Twitter API Investigation

If you want to develop a Twitter application and use Twitter API, you should register your application to the Twitter developer center. All queries should be done with a Twitter account, which means anonymous queries are not accepted. Twitter has three different API calls, Search API, REST API and Stream API [31]. Search API

is developed for searching keywords and people according to search criteria. REST API can be used for gathering interested user profile and tweet history, and lastly Stream API can be used for collecting data dynamically. Search API and REST API has query limits which means you can use limited number of query. Query limits are determined according to query types. You can find detailed information at [32]. In our thesis, we are interested in selected user profile, tweets and followee/followers therefore we use Twitter REST API.

# CHAPTER 3

# USER PROFILING

In this chapter, we will explain the first part of the proposed solution: User Profiling. Firstly, we will explain the most important related works about information extraction. After that, we will explain the details of user profiling sub-part.

## 3.1 Summary of Related Works about Information Extraction from Microblogs

Information extraction from microblogs has been a heavily studied research problem. We particularly focus on the studies about extracting the sentiment data or user preference extracting studies in microblog domain.

Extraction of useful and structured data from microblogs has been a popular topic in recent years [33]. In [34], authors develop an algorithm to overcome data sparsity problem in short texts. In [35], various Twitter influence measures in the literature are collected and classified. [7], [36] and [2] propose solutions to extract user inner preferences and latent attributes by using microblogs. [37] proposes a short text classification system. [23] proposes a framework for information collection. Their work mainly focuses on Social Network Analysis (SNA) and graph theory by using groups of people and their characteristics. In [25], authors propose the algorithm called TextRank for processing texts. [3] describes a solution for extracting annotations from user microblog. Authors propose a system that finds the similarity of a text and Wikipedia concepts in [30]. [8] presents a survey about Twitter summarization algorithms and they improved TF-IDF algorithm performance by adding hybrid property into this algorithms.

Sentiment analysis and public opinion analysis of a topic in microblogs is another hot research area in recent years. In [38], a novel clustering method is proposed for sentiment analysis on the Twitter dataset. In [39], authors develop a lexicon based approach for sentiment analysis on Twitter, namely SentiCircles. In [26], authors propose a system that extracts the implicit sentiment data for social media marketing. They develop an emotion categorization model, named Hourglass. In [40], a word of mouth quality classification is applied and a sentiment lexicon is generated from the contextual information. In [41], authors investigate the effects of emotional expressions in the electronic word of mouth. In [6], authors propose a framework to analyze the opinion groups in blogs as positive and negative. [28] develops a project to calculate quantitative opinion score (positive and negative) about a topic in Microblogs. [4] develops a solution for determining public opinion about interested topic by using microblogs.

These studies reveal the importance of sentiment for user profiling in microblogs. In an earlier work [21], we propose a sentiment enhanced hybrid TF-IDF method for extracting user profile in Twitter.

## 3.2 User Profiling (Keyword Extraction)

In this section, user profiling part of our thesis will be introduced. At the end of this part, user profiling keywords and their weights will be calculated and these data will be used to match with advertisement data. Keyword extraction can be divided into two subsections; proprocessing step and data extraction step.

### 3.2.1 Preprocessing Step

This section is mandatory for extracting the meaningful data from microblogs, because microblogs are noisy and ungrammatical text blocks.

1. Link handling: Preprocessing starts with link handling; we ignore link data and remove the hyperlinks from tweets. Although, link information is retained in some of the related studies, in our approach we remove it since we aim to

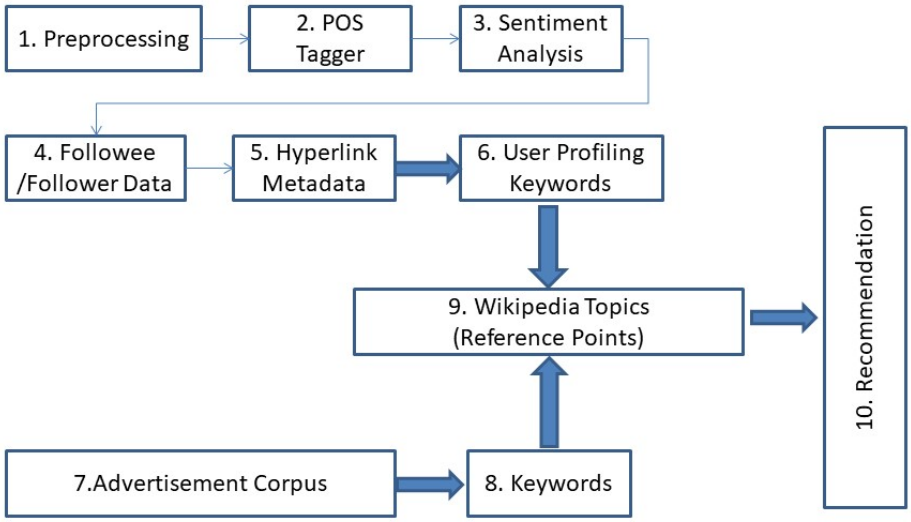



Figure 3.1: Overview of the Proposed Method

keep the scope within the text ignoring the external resources. Our Web Spider module collects the text information from landing page of the link but actual link string is ignored.

2. Metadata pruning: Retweets' meta information is filtered, because "@Proper Noun" combination does not contain any useful information for user profiling. This info is used for calculating the interaction count between the main user and this friend.

3. # char removal: Only # char is removed from hashtags because in Twitter this symbol is used for representing the start point of hashtags. Hashtags are important for information extraction, therefore they are not filtered. Moreover, hashtags are processed as nouns.

4. Slang words: Internet slang words are composed of abbreviations and sometimes improper words for user profiling, therefore in our preprocessing part, internet slang words are replaced with their original words and some improper words are pruned. We have a look-up table for this operation and this table

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBP want)
      (S
        (VP (TO to)
          (VP (VB eat)
            (NP (JJ Mexican) (NN pizza) (NNS ASAP)))))))))
```

Figure 3.2: POS Tagger result without Slang Word preprocessing

consists from the most popular internet slang words. Slang words are replaced by full length text by looking this table.

5. POS Tagger: Stanford POS Tagger ([42]) is used for extracting nouns in user tweets. We think that, we can categorize user interests and user profiles by using only nouns. More part of speech labels can give better results but in our system we believe that nouns are the most adequate label types. Therefore, other part of speech words are ignored for our system. Only nouns are extracted and these words are labeled as potential keywords for user profiling. Stanford POS tagger noun labels are:

   - NN:Noun,singular or mass
   - NNS:Noun plural
   - NNP:Proper noun singular
   - NNPS:Proper noun plural

   POS taggers give good results when sentence is regular and formatted; on the other hand if the text is noisy, POS taggers sometimes produce incorrect results. For example, for the sentence "I want to eat Mexican pizza ASAP"; Stanford POS tagger result can be seen in Figure 3.2. In this sentence "ASAP" is a slang word which is an abbreviation of "As Soon As Possible". Proposed algorithm takes only nouns for information extraction from user account. According to tagger labels the word "Pizza" is NN and the word "ASAP" is NNS. It can easily seen that "ASAP" is labeled incorrectly. Therefore, this word should not be used as extracted keyword for user profiling. In order to solve this problem, proposed work removes slang words in the previous step of preprocessing and correct result can be obtained as shown in Figure 3.3.

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBP want)
      (S
        (VP (TO to)
          (VP (VB eat)
            (NP (JJ Mexican) (NN pizza))
            (ADVP
              (ADVP (RB as) (RB soon))
              (PP (IN as)
                (ADJP (JJ possible)))))))))
```

Figure 3.3: POS Tagger result with Slang Word preprocessing

```
(ROOT
  (S
    (NP (NNP Yes) (NN man) (NNP Golf))
    (VP (VBZ is)
      (NP (DT the) (JJS best) (NN car)))))
```

Figure 3.4: POS Tagger result without character count postprocessing

As described, several preprocessing steps are applied in order to transform noisy texts into regular text. However, due to the nature of the microblogs, they may not prevent all the problems arising from informal language. Therefore, Stanford POS tagger results should be post-processed in order to improve the results. According to our observations, POS tagger results with incorrect tagging mostly for short words. For example, for the sentence "Yes man Golf is the best car.", whose POS Tagging is shown in Figure 3.4. In this sentence, "man" is a 3 character noun but we know that it is not a relevant word for user profiling. Therefore, we ignore one and two character length words. In addition, we prepare a look-up table which consists from common three character length English words. Three character length nouns are compared with this table and according to match result our system accepts this noun or ignore them. Longer nouns are accepted directly. According to our development observation, this post-processing operation increases algorithm's performance.

6. Punctuation Marks: Our system filters punctuation marks in order to clear the words for POS tagging.

### 3.2.2 Data Extraction Step

There are several methods to find the weight, and hence the importance of the words appearing in a text. The most popular methods in the literature are TF-IDF and PageRank algorithms. According to TF-IDF definition; a word is important for a document, if frequency of this word is high in this document but frequency of this word is low in document collection. It means, this is an important word for this document. By this way, stop-words can be eliminated, since stop-words frequencies are high for all documents. On the other hand, TF-IDF algorithm has a problem in microblog domain for the question "how to define the document collection". If we process all tweets of a specific user as a document and we start TF-IDF algorithm; we can only calculate TF value. IDF functionality will be ignored. If we select all tweets as different documents, we will have a document collection. However, this will reduce the effect of TF value almost to null. A tweet has character limit, document frequencies will be very low and algorithm gives misleading results. To solve this problem, the study in [3] uses a different solution; interested topic is selected, according to this topic they run a Twitter query. Randomly selected N users' tweets are used as a document collection. A user's all tweets represent a document.

In another study [8], a hybrid version of TF-IDF algorithm is proposed. According to this algorithm, all tweets are processed as different documents. Moreover, the main user's all tweets represent a document collection except for one property; when computing the word frequencies, all occurrences of all tweets are used. By this way, term frequencies have normal values and IDF property is not lost. Related calculations are given in Formula 31, Formula 32 and Formula 33. In the literature, Hybrid TF-IDF has one of the best performance in this domain [8]. Therefore, we choose this algorithm as basis for our study.

$$W(w_i) = tf(w_i) * log_2(idf(w_i)) \tag{31}$$

$$tf(w_i) = \frac{\#OccurrencesOfWordsInAllPosts}{\#WordsInAllPosts} \tag{32}$$

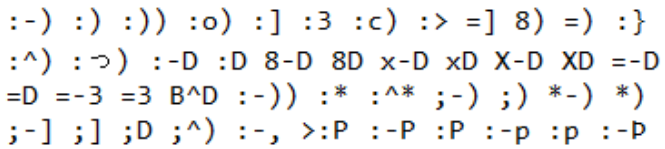



Figure 3.5: Positive Emoticons

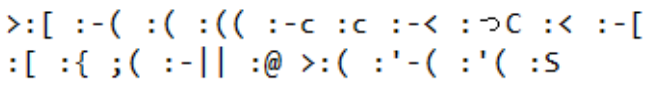



Figure 3.6: Negative Emoticons

$$idf(w_i) = \frac{\#Posts}{\#PostsInWhichWordOccurs} \tag{33}$$

We extend this method, resulting in a new TF-IDF calculation method: sentiment based hybrid TF-IDF calculation. Sentiment analysis feature is incorporated into the method by using emoticon list and subjective keywords. We have two different emoticon list: positive (Figure 3.5) and negative emoticons (Figure 3.6). Moreover we have a subjectivity list with this format: $type = weaksubjlen = 1word1 = goodpos1 = anyposstemmed1 = npriorpolarity = positive$ [43]. By using emoticons and subjectivity keywords we decide a coefficient value about a tweet. This coefficient can be positive or negative. We compare all words of a tweet with these look up tables. All emoticons and subjectivity words have a sentiment value. For example, a positive emoticon value is +3, on the other hand a negative emoticon value is -3. Subjectivity words values are calculated by strong-weak subjectivity and positive/negative properties. For each tweet, all positive values and all negative values of words and emoticons are extracted. As a result, for each tweet an overall sentiment score is calculated.

For example, if the sentiment score of a tweet is 2, TF value for each word in this tweet is increased by this value $(1 + 2 = 3)$. If sentiment value is negative (for example -1), each count of this tweet words frequency value is decreased by this value $(1 + (-1) = 0)$ for hybrid TF-IDF calculation. To sum up, if a user writes a positive tweet about a topic, TF-IDF values of these keywords are boosted, on the other hand if a user writes negative tweet about a topic, then the importance of this

tweet is reduced. As the final step, according to TF-IDF values, top N keywords are selected as user interest topics.

### 3.3 Enhanced User Profiling (User Model)

Since the length of a microblog post is limited, users extend their text messages with hyperlinks and media items such as photos and videos. In this study, we enhance user profiling performance by using hyperlink data. The proposed system explores the linked web pages and collects web page metadata. For *Youtube*[1] and *Instagram*[2], which are the most popular hyperlinks for Twitter, web spider collects title and description of web pages. For other sites, only title information is collected and replaced with link string in the related tweet. By this way, hyperlink web data is exploited to increase information extraction performance.

Additionally, user's followee and follower lists are useful to extract user's passive preferences. In this study, we use both followee and follower lists to enhance user profiling performance (Figure 3.1 Step 4). In user profiling part, the proposed system collects user tweets and extracts top *n* (default value 5) keywords by using sentiment enhanced hybrid TF-IDF model. To enhance profiling capability, main user's followee and follower lists are collected and top *n* keywords are extracted from each of these users. To create a user influence model, the following three important rules are applied:

1. The main user's interests should be used as core keywords: Keywords from followees/followers may distort the focus of main user's interests. Therefore, proposed system selects keywords from the main user's tweets, and followee/ follower's keywords only increase the TF-IDF values of the main user's keywords.

2. The numbers of Followees/Followers should be normalized: If $n$ followers are used in influence model, each follower has $\frac{1\times\alpha}{n}$ effect on TF-IDF weighting. $\alpha$ coefficient represents the importance value of a follower. In this equation, $\alpha$'s

[1] www.youtube.com
[2] www.instagram.com

value is in the range of [0, 1], where 0 means that the follower has no importance, and 1 means that the follower has the equal importance with the main user. In our experiments, we used the equal $\alpha$ value for all followers. Similar to $\alpha$, $\beta$ value represents the importance factor of followees and it is in the range of[0, 1]. $\alpha$ and $\beta$ coefficients are independent variables and they do not affect each other. If $n$ followees are used in influence model, similarly, each followee has $\frac{1\times\beta}{n}$ effect on TF-IDF weighting. The coefficients $\alpha$ and $\beta$ denote that main user's interests are more important than followees/followers. Furthermore, different coefficients indicate that followers and followees have different weights. Generally, followees' keywords are considered more effective than followers' keywords.

3. Interaction quantity should be considered: In the followee/follower list, some users are more important than the others, due to the number of interactions. It is assumed that more interaction represents more common interests. In our work, we formulate this importance level ($\#Interaction(i)$ value in Equation 35 and Equation 36) by counting the retweets of the main user from followees and retweets of followers from the main user, where the default value is 1.

For a given keyword, the overall score is calculated as given in Equation 34. The first component of the equation gives the score of the keyword through user's tweets by using the method described in Section 3.2 and its extension with analysis of media content. The second component, described in Equation 35, denotes the score from the followers, and the third component, described in Equation 36, denotes the score from the followees. In our experiments, we set $\alpha = 0.2$ and $\beta = 0.3$, which have been determined experimentally.

$$TFIDF(K) = UserTFIDF(K) + FolloweeModel(K) + FollowerModel(K) \tag{34}$$

$$FollowerModel(K) = \sum_{follower\ i=1}^{n} \#Interaction(i) \times \frac{1 \times \alpha}{n} \times TFIDF(i)(K) \tag{35}$$

$$FolloweeModel(K) = \sum_{followee\ i=1}^{n} \#Interaction(i) \times \frac{1 \times \beta}{n} \times TFIDF(i)(K) \tag{36}$$

# CHAPTER 4

# ADVERTISEMENT RECOMMENDATION

In this chapter, we will explain the second part of the proposed solution: Advertisement Recommendation. Firstly, we will explain previous studies which are about recommendation and advertising. After that, we will explain the details of the advertisement recommendation sub-part.

## 4.1 General Information about Textual Advertising

Internet advertising is the fastest growing marketing model nowadays. It is a hot topic in marketing area because it grows very fast and it has very large volume. According to IAB/PwC Digital Advertising Revenue Report [44]; in 2014 annual US interactive ad revenues broke $49 billion, marking the fifth consecutive year of double-digit annual growth. 16% (or $6.7 billion) increase from 2013's $42.8 billion. The growth of the advertising types for the last five years are; Internet 135%, Mobile 110%, Broadcast TV 79% and Cable TV 69% [44]. Textual advertising is the largest part of internet advertising. The main problem in this area is finding the correct advertisement set. If selected advertisement is related to user interests, revenue will be increased. On the contrary, irrelevant advertisements annoy users. Therefore, advertisement selection is the most important part of textual advertising. Intelligent algorithm development is the hot topic in this area. Textual advertising can be divided into two forms.

1. Sponsored Search: If the user search a query in the search engine, there will be some advertisement blocks in the search engine result page. These advertisements are related to the user interests and they are selected after analysis of user query log or single query keywords.

2. Contextual Advertising: It is the generic version of the sponsored search. In this model, advertisement block can be seen in any kind of web page instead of search engine result page. Advertisements are selected after analysis of the host web page's contents. For contextual advertisement, there are four different roles.

    - Advertisement Owner: This is the money supplier role of the system. Mainly, this is a company which wants to announce a product or a company name by an advertisement. They have landing pages, when user clicks advertisement icon, user will go to the company web page.
    - Web Page Owner: This is the host web page. Advertisements are located in this web page. When user clicks the advertisement, this role earns money.
    - Platform Supplier: This is the owner of the infrastructure. It has advertisement database and it knows the web pages contents. The system selects advertisements according the web pages' contents. This role supplies infrastructure to the web page owners to publish these advertisements. First Contextual Advertisement system is supplied by Google. This role shares revenue with Web Page Owner.
    - User: A person who visits the web pages. If this person buys something from advertiser, advertisement process is successfully completed.

Moreover there are lots of payment types such as pay-per-click, pay-per-action, pay-per-impression. Broder [45] and Pak [46] define the estimation of revenue for a page:

$$R = \sum_{i=1..k} P(click|p, a_i) price(a_i)$$

in this formula k = number of displayed advertisement in the page p. To simplify model, previous studies ignore the pricing model. Therefore, formula is simplified as total number of click count.

$$R = argmax P(click|p, a_i)$$

In contextual advertising area, previous studies mainly deal with syntactic keyword matching between target web page and advertisement description. This approach has

some problems, therefore last studies concentrate on the enrichment operation for solving these problems ([46] and [47]):

- Homonomy and Polysemy: "Rice" keyword can be used for "Condoleeza Rice" or a grain product. It causes semantic ambiguity.
- Intersection problem or vocabulary impedance: It happens when a page or an advertisement is represented by limited number of keywords. For example a web page mainly use keyword vehicle but advertisement can be represented as automobile. Semantically they have same meaning but as a result of the vocabulary impedance problem they are mismatching.
- Context Mismatch: Sometimes keywords match but semantically there is a problem in this matching. For example, if we show tourism in China advertisement in a news page which is about natural disaster in China, it is a mistake.

## 4.2 Related Works about Contextual Advertising with Wikipedia

In this section, a summary of related studies that address advertisement recommendation by using Wikipedia are given. In the literature, advertisement or product recommendation problems have been heavily studied. In this section, the studies that involve microblogs or Wikipedia are summarized.

DBPedia is the structured version of Wikipedia. For each Wikipedia page, a Uniform Resource Identifier (URI) is created at DBPedia knowledge base ([48]). This knowledge base contains textual descriptions; titles, abstracts, info-box related data, category information and page links. At 2018, the English version of the DBpedia knowledge base describes 4.58 million things, out of which 4.22 million are classified in a consistent ontology and DBPedia provides localized versions in 125 different languages. One important property which increase DBPedia usage is SPARQL language support. User can use SPARQL queries for extracting data from DBPedia ontology. To sum up, DBPedia is the multilingual structured version of Wikipedia, therefore it is a free knowledge base.

Zongda Wu *et all.* [49] proposed a system which is mainly deal with the advertise-

ment matching for target web page by using Wikipedia concept, category and target web page keyword properties. They claim that only keyword usage for advertisement matching has some problems like; homonomy, polysemy and low intersection of keywords. Therefore, they add Wikipedia category and concept data in order to enrich target web page semantic data. In their work, the web page and advertisements are defined by three different vectors. Keyword vector, concept vector and category vector. For matching process; they use cosine similarity between these vectors. They use intelligent algorithms to decrease calculation costs. For example, in an offline process they select concepts which are related to their advertisement set.

Dorothea Tsatsou *et all.* [50] proposed an advertisement recommendation system which combines ontological knowledge with content extracted linguistic information. According to their work, there are two problems for recommendation systems; word impedance problem and cold start problems. Word impedance problem can be solved by extracting semantic data from textual content and describing this content in machine understandable format. For classifying textual content; they use domain ontology to convert textual data to the machine understandable classes. Their system analyze the textual data (advertisement, annotated video or short texts) when user consumes this item. They prepare a domain ontology in an offline process and extracts semantic data from user consumed items. Semantic user profile matches with domain ontology by looking text similarity of keywords and concepts. Likewise, advertisements are matched with domain ontology terms. That means, domain ontology nodes are the reference points between page profile and advertisements. Their system has an disadvantage that for each domain they should prepare a domain ontology. In their paper, they show the evaluation results on the soccer domain with a soccer ontology.

Weinan Zhang *et all.* [51] mainly deal with the problem about advertisement recommendation for short text pages. Short pages contain insufficient textual information for ranking therefore classical ranking-based algorithms have low performance for these kind of web pages. In order to overcome this problem; they proposed a system which enriches the target web pages by new keywords. These keywords are not located in the target web page but they are relevant with the web page. Their algorithm consists from following steps: 1. Web page is analyzed and all Wikipedia entities

that occur in the page are extracted. By using SVN approach, all these entities are scored as content relevant score. 2. Each entity is scored according to occurrences of this entity in the predefined set of advertisement. This advertisement relevant score calculation can be done in offline process. As a result, each Wikipedia entities has two score; content relevant score and advertisement relevant score. 3. With these two scores, they define PageRank algorithm iterations. Lastly, they select highest k entities as the recommendation keywords. In evaluation part of their work; they use human judges. Page and keyword tuples are voted as they are related or not related (1 or 0).

Broder *et all.* [45] clearly define the differences between Contextual Advertising and Sponsored Search. Contextual advertising means placement of an advertisement or an advertisement block which are selected according to the target web page content on a generic web page. On the other hand, sponsored search represents the placement of advertisements according to user query on the query result page. They mainly deal with the contextual advertising approach. They add semantic data to the syntactic data in order to increase the contextual advertising quality. On the other hand, without semantic profiling; only keyword matching approaches have some problems. For example golf player "John Maytag" can be matched with popular Maytag dishwashers but they are different topics. Therefore, semantic approach should be used to find correct domain. In order to match advertisement and page, they develop a taxonomy tree with 6000 nodes. This is a commercial taxonomy which is built by Yahoo corp. Taxonomy is created for classifying commercial interest queries and it is populated by human editors. In this work; pages and advertisements are represented with the combination of the keyword vector and taxonomy node relevance vector. Each advertisement and page is analyzed to find the relevance score according to the vector of taxonomy tree nodes. For matching between advertisement vector and target page vector; cosine angle is used. Moreover, they have a domain taxonomy and they know user preferences therefore they can apply diversification operation by using tree relationship.

Ribeiro Neto *et all.* [47] proposed impedance coupling strategy for vocabulary impedance problem. They define this problem as low intersection of keywords even they are related to each other. To solve this problem, they develop an enrichment strategy called

impedance coupling. This strategy is adding new pages into the system which are related to the targeting (triggering) web page. New keywords are extracted from these new web pages and added to the triggering web page keywords. They use Bayesian Network to show adding new keywords will improve the characterizing the main topic. By that way, they enrich triggering web page contents and they decrease the vocabularity impedance problem probability. To show the improvement, they use five different algorithms before impedance coupling and five new strategies for impedance coupling. They use 93.972 advertisement, 100 triggering web pages and 5.939.061 related web pages for evaluation. Human judges (group of 15 user) are used for evaluation results. They gain nearly %60 of improvement for pages and advertisement matching.

Pu Wang *et all.* [52] deal with the document classification problem. According to their study, classic document classification methods are based on keyword representation of the syntactic approach, semantic meaning is not used. Some previous studies add semantic approaches but they have limited coverage capability. Therefore, in their work they use the biggest data set Wikipedia for document classifications. Wikipedia is the largest encyclopedia but it has some problems; unstructured and noisy. They develop solutions for synonymy, polysemy, hyponymy and associative relations problem. At the end, they create a Wikipedia thesaurus and they enrich target document by Wikipedia content. By that way they increase document classification performance.

Pak *et all.* [46] define the problem about contextual advertising with simple keyword matching technique. They claim that this technique has poor accuracy because of homonomy, polysemy, low intersection of keywords and context mismatch. Therefore, they propose a method for increase the performance of contextual advertising. They use Wikipedia articles as the reference points between target page's content and advertisements. For evaluation part they use 100 news pages. They select 1000 Wikipedia articles as the reference points and 7996 advertisements to match. Each target page is represented by the keyword vector. By using cosine angle, the similarity between each page and each article is calculated. Same process is done between advertisements and articles. They have two different methods for calculating overall similarity between page and advertisement; Wikipedia similarity and Wikipedia distance. As a result they claim that they have %50 lift in the average precision.

Saul Vargas *et all.* [53] proposed a framework to increase recommendation quality by using diversification of user sub-profiles. They claim that, people have different preferences and different attitudes in different contexts such as; sports, politics, work, leisure, music or movies. For example, user classical music preferences can be more valuable than heavy metal music preferences in order to recommend similar songs. If user profiling is divided into sub domains they claim that recommendation quality will be increased. Moreover, they proposed that recommendation without diversification is not complete for user benefits. User always see the similar items because of the recommendation accuracy and get limited benefit from this recommendation. Therefore user profile diversification shall increase the recommendation quality.

The main logic for diversification based recommendation systems is implementing the greedy selection algorithm. In this algorithm, main operation is maximizing the $f_{obj}$ (objective function) which embodies the accuracy-diversity balance. This balance increase the recommendation quality because recommended items are user related but not user consumed items. In their approach, user profile is partitioned according to item domain. Recommendation items are selected by looking these sub profiles.

Wu *et all.* [10] proposed a work which improves Pak's [46] solution. They define a new terminology in this field: Selective Matching. Wu and his friends use same algorithms with Pak's work, but they change the reference Wikipedia papers selection algorithm. In Pak's work, he uses all Wikipedia featured articles. On the other hand, Wu uses four different methods to select reference pages: Complete-title-matching method, All-keywords-matching method, Any-keyword-matching method and lastly All-article-matching pages. All-article-matching pages represent all featured articles which are used in the previous studies. Other selections represent different subsets of this big collection. In this study, they have achieved previous study relevance level (Figure 4.1) with better running time performance (Figure 4.2).

To sum up, the recommendation studies within microblog context generally focus on product recommendation or other type of recommendation items such as news or user. In [54] and [55], authors develop a system called METIS, which recommends products according to user interests. They use Sina Weibo [1] as the data source. In [56],

[1] Sina Weibo is the Chinese microblog with the highest number of users. www.weibo.com

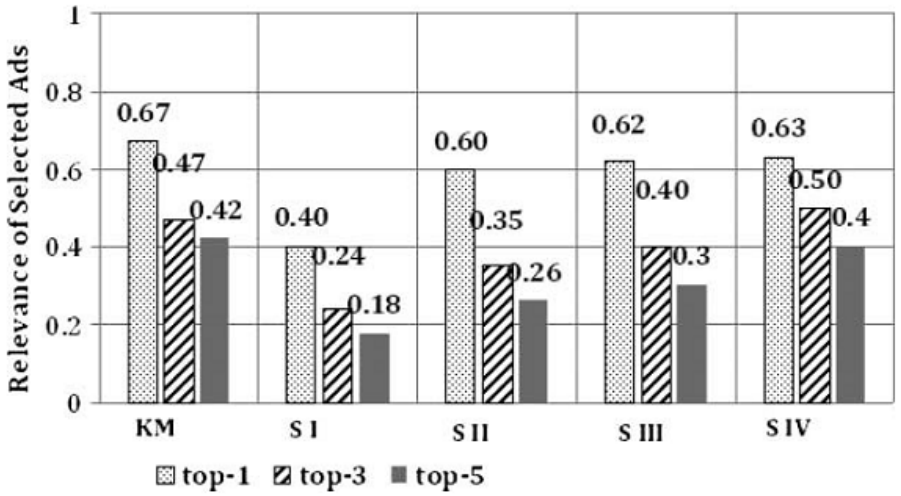



Figure 4.1: Average Relevance of Textual Ads [10]

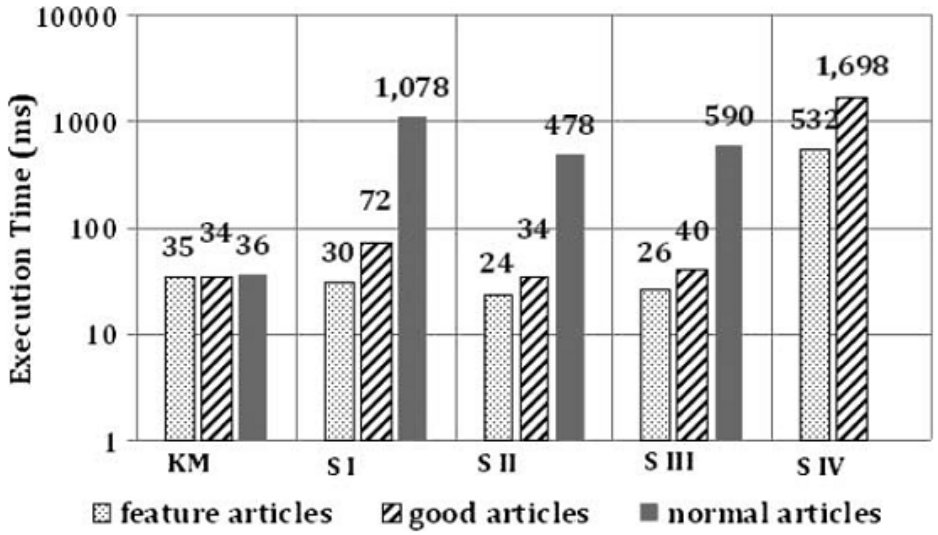



Figure 4.2: Running Performance [10]

location data is also included in order to make more accurate recommendations. In [57], real time news/user recommendation is generated to the user. The study in [58] proposes a model to recommend personalized tweets. In [5], a system to spread advertisement by using microblog is proposed. Their work differs from existing works because they use microblog as the data source rather than traditional websites [59]. The work in [24] presents a solution to understand user preference to assist marketers for target marketing. They use *Plurk*, which is a popular microblog in Thailand. The study in [29] proposes a system for microblogs that reviews the products from five different categories: applications, music, movies, books and games, and recommend products to the users.

There is a vast number of studies on advertisement recommendation for web pages. The work in [50] defines an advertisement recommendation system that combines ontological knowledge with content extracted linguistic information. In [51] , it is mainly dealt with the problem of advertisement recommendation for short text pages. The authors propose a system that enriches the target web pages by new keywords that are relevant with the web page. The work [45] defines the differences between contextual advertising and sponsored search, and the authors focus on the contextual advertising approach. The authors extend syntactic data with semantic features in order to increase the contextual advertising quality. In [47], an impedance coupling strategy is proposed for vocabulary impedance problem. This strategy involves adding new pages that are related to the targeting (triggering) web page into the advertisement system.

Wikipedia usage is another popular topic for solving the above-mentioned problems by enriching the original document. In [60] and [61], Wikipedia category structure is used for solving the shortcomings of existing approaches of similarity computation. In [62], Wikipedia concept vector is used to understand the text semantics and to improve the accuracy of classification. In [63], authors use Wikipedia to convert concept based representations of documents from one language to another to generate cross language text classifier. In [64], scientific documents are classified by using their titles with the help of Wikipedia enrichment. In [65], a survey study is presented on Wikipedia corpus usage in such problems. The work in [62] proposes a new classification approach that represents a document as a concept vector in the Wikipedia

semantic space. In [60], a new solution is proposed for calculating the semantic similarity between concepts. The authors use Wikipedia category structure (large directed acyclic graph). The studies in [49] and [66] propose a system that matches advertisement with target web page by using Wikipedia concept, category and target web page keyword properties. The authors devise algorithms to decrease calculation costs. In [52], authors use the largest data set of Wikipedia for document classification. The study in [46] focuses on the problem of contextual advertising with simple keyword matching. The authors use Wikipedia articles as the reference point between target page content and advertisement. In [10], authors propose a method which improves the solution described in [46]. The authors use the same algorithms with [46]; but in their study, Wikipedia papers reference selection algorithm is changed.

The proposed method in this work uses Wikipedia pages, however, there are two major differences from the previous studies that use Wikipedia pages for advertisement recommendation. Firstly, this study differs from the previous studies, which match advertisements and websites, since we focus on microblog as the advertisement media, hence the matching is applied between advertisement and microblog user. Hence the elements to be matched are not the same as in the previous studies, since the user is represented with a limited set of keywords. Additionally, this difference needs a different processing for keyword extraction from microblog account in comparison to web pages. The keyword extraction process is documented in detail in [21]. Secondly, previous studies, which match an advertisement with a web page, use Wikipedia Featured Articles, Wikipedia Titles or Wikipedia link hierarchy. The proposed study uses Wikipedia Good articles since it contains more information. Using a larger set may affect the performance leading to longer running times, but the proposed system solves performance problem and an experiment on this analysis in the experiment section.

In the evaluation part of this article, the recommendation accuracy of the proposed method is compared with Keyword Matching method, the method described in [46] and the method described in [47]. Since these studies are the most referenced studies in their area, they are selected for comparison. The results show that, the proposed method has a higher accuracy in comparison to these baseline studies. The similar results are also given in the article [46], which proposes a Wikipedia based similar-

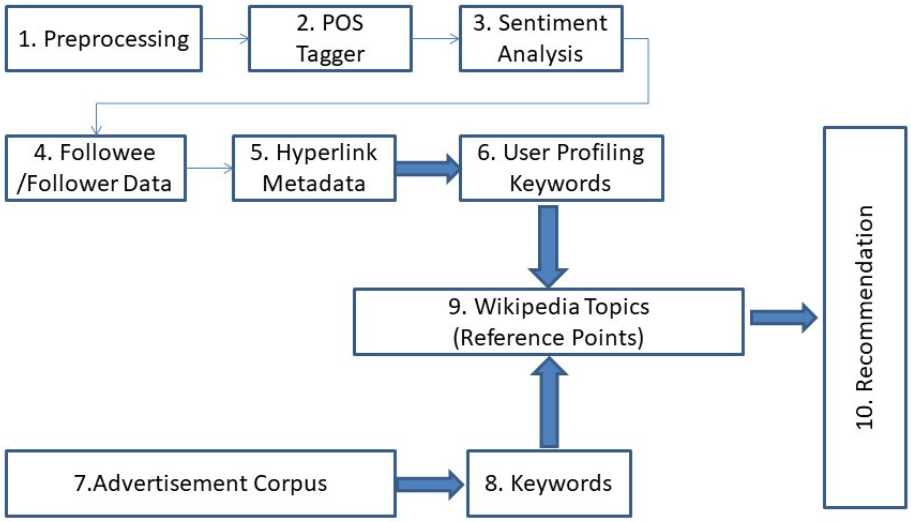



Figure 4.3: Overview of the Proposed Method

ity method, claiming that their study has better recommendation accuracy compared to [45]. Indirectly, it can be inferred that the proposed solution has a better recommendation accuracy in comparison to [45], which employs a taxonomy to increase recommendation quality.

## 4.3 Proposed Method

The proposed method is composed of three sub-phases: User interest model construction, advertisement model construction and matching these two models to recommend the best advertisement set. The overview of the proposed solution is presented in Figure 5.1. In the first subsection, basic user profiling method, which has been proposed in [21] and Chapter 3, is summarized. In the second subsection, advertisement matching and recommendation steps are explained.

### 4.3.1 User Profiling

The first six steps in Figure 5.1 shows the user profiling phase of the proposed method. The first three steps given in the figure constitute the basic user profiling model, which is described in [21] and Chapter 3. The summary of these six steps is as follows:

1. Preprocessing: Microblogs are noisy text blogs, therefore user profiling phase includes a preprocessing step to eliminate unnecessary terms as follows:
   (a) Link handling: Hyperlinks are removed from the tweet contents and only user generated texts are used in main user modeling.
   (b) Metadata pruning: Retweets' meta information is filtered, because "@Proper Noun" combination is ignored.
   (c) # char removal: Only # char is removed from hashtags, but hashtags are retained.
   (d) Slang words removal: Slang words and abbreviations are replaced with their original words. A look-up table is used to replace by full length text.
2. Part of Speech (POS) Tagging: Nouns are extracted by using Stanford POS tagger[42] and these words are labeled as candidate keywords for user profiling.
3. Sentiment Analysis: The basic user profiling involves sentiment based hybrid TF-IDF calculation, which is proposed in [21]. This model represents the user interests by top-5 keywords having positive sentiment.
4. Followee/Follower Data: Followees and Followers of the main user are collected and their keywords are used according to interaction count between main user and the friend.
5. Hyperlink Metadata: Web spider collects the keywords from link page and our system adds these keywords into the main user keywords.

### 4.3.2 Advertisement Recommendation

In the literature, a common textual advertisement has four fields: Title, Body, URL, and Bid-Phrases. Among these parts, *title* is the visible part of an advertisement in the

host web site, *URL* is the advertisement landing web site, *body part* is the textual part of the advertisement, and *bid-phrases* section is keywords of the advertisement. For advertisement matching, our method uses bid-phrases since they are the summarized version of the advertisement body.

As shown in Figure 5.1 before recommendation, system generates keywords. These keywords are; user profiling keywords (In Step 6), and advertisement corpus' keywords (In Step 8). As discussed in related works, without any reference points, keyword matching quality is not satisfactory enough. Therefore, we need reference points to match advertisements and the user profile. To overcome this problem, we use *Wikipedia*.

Wikipedia has different page types. Most structured and well defined wiki-pages are called as *Wikipedia Featured Articles*. Several previous studies use *Wikipedia Featured Articles* set in their studies (such as [46]). They are well defined and structured but Featured Articles are too few to cover a wide range of domains. According to statistics from [67], there are 5,463,593 articles on Wikipedia, only 5,113 are listed as featured articles (about 1 in 1,070) and 26,424 are categorized as good articles (about 1 in 207).

In this work, we aim to develop a domain independent solution. Hence, in order to include a rich enough set of domains, we use Wikipedia Good Articles instead of Wikipedia Featured Articles. Wikipedia Good Articles set is the largest structured subset of the Wikipedia articles. It means that Wikipedia Good Articles set contains more data in comparison to Wikipedia Featured Articles. We preferred to use the larger set over Wikipedia Featured Articles considering that it provides a higher capacity for matching. *Wikipedia Good Articles* are well structured and it has enough number of pages to define reference web pages. A Wikipedia Good Article meets the good article criteria defined in [68]: Well written, verifiable with no original research, broad in its coverage, neutral, stable, illustrated. We use 22,900 Wikipedia good articles (all Wikipedia Good Articles available as of December 2016). We collect all texts (title + body) from these pages and they are used to calculate corresponding term vector.

Firstly, user profile, Wikipedia pages and advertisements in the corpus are represented

as keyword sets. Equation 41 shows the keyword set of a given Twitter user $tk$, where each $tk_j$ is a keyword in the user profile, which is extracted from Twitter posts.

$$tk = \{tk_1, tk_2, tk_3, ...tk_n\} \tag{41}$$

In Equation 42, keyword set of a given advertisement $ak^i$ is given. In this representation, each $ak^i_j$ is a keyword extracted from advertisement content. $ak^i_j$ represents the $j^{th}$ keyword of advertisement i.

$$ak^i = \{ak^i_1, ak^i_2, ak^i_3, ...ak^i_m\} \tag{42}$$

In Equation 43 corresponding keyword set of a given Wikipedia page $wk^i$ is given. In this representation, each $wk^i_j$ is a keyword extracted from Wikipedia page. $wk^i_j$ represents the $j^{th}$ keyword of Wikipedia page i.

$$wk^i = \{wk^i_1, wk^i_2, wk^i_3, ...wk^i_p\} \tag{43}$$

Once keyword sets are available, TF-IDF values of the keywords are calculated. For Wikipedia pages' keywords, well-known TF-IDF formula given in Equation 44 is used. Equation 45 calculates the Term Frequency (TF) value of keyword *j* for Wikipedia Page *i*. In this equation, $OK_jWP^i$ stands for Occurrences of Keyword j in Wikipedia Page i and $WWP^i$ stands for Words in Wikipedia Page i. Equation 46 calculates the Inverse Document Frequency (IDF) of keyword *j*. In this equation, $WGP$ stands for Wikipedia Good Pages and $WPK_jO$ stands for Wikipedia Pages in which Keyword j Occurs.

$$TFIDF(wk^i_j) = tf(wk^i_j) \times idf(log_2(wk^i_j)) \tag{44}$$

$$tf(wk^i_j) = \frac{\#OK_jWP^i}{\#WWP^i} \tag{45}$$

$$idf(wk^i_j) = log_2(\frac{\#WGP}{\#WPK_jO}) \tag{46}$$

For Advertisements' keywords, similar to Wikipedia keywords, TF-IDF formula given in Equation 47 is used. Equation 48 calculates the TF value of keyword *j* for Advertisement *i*. In this equation, $OK_jA^i$ stands for Occurrences of Keyword j in Advertisement i and $WA^i$ stands for Words in Advertisement i. Equation 49 calculates the

IDF of keyword *j*. In this equation $TA$ stands for Total Advertisements and $AK_jO$ stands for Advertisement in which Keyword j Occurs.

$$TFIDF(ak_j^i) = tf(ak_j^i) \times idf(log_2(ak_j^i)) \tag{47}$$

$$tf(ak_j^i) = \frac{\#OK_jA^i}{\#WA^i} \tag{48}$$

$$idf(ak_j^i) = log_2(\frac{\#TA}{\#AK_jO}) \tag{49}$$

Once TF-IDF values are calculated, corresponding keyword vectors are constructed for the user, advertisements and Wikipedia pages. As given in Equation 410, similarity values between two keyword vectors are calculated by *cosine similarity*. In this equation, $a^j$ represents advertisement *j* keyword vector and $w^i$ represents Wikipedia Page *i* keyword vector.

$$sim(a^j, w^i) = \frac{a^j.w^i}{|a^j|\,|w^i|} \tag{410}$$

The Similarity between each advertisement and each Wikipedia page is calculated in order to obtain advertisement-Wikipedia page similarity matrix. In this work, we use a different and novel normalization process. For each advertisement, Wikipedia pages are sorted according to similarity values. Each advertisement is represented by top-most similar 100 Wikipedia pages. In this manner, we eliminate the advertisement domination factor. Some advertisements' keywords are included in most of the Wikipedia pages. To reduce this boost effect, we represent each advertisement by top-most similar 100 Wikipedia pages.

As in the previous steps, similarity between the user and Wikipedia page is calculated by using cosine similarity. Lastly, to calculate similarity between the user and an advertisement, vectors' dot product is used as given in Equation 411. In this equation, $t$ represents the user keyword vector, $a^j$ represents keyword vector of Advertisement *j*, and $w^i$ represents keyword vector of Wikipedia Good Pages *i*. Appendix A shows the intermediate documents of this process.

$$sim(t, a^j) = \sum_{\forall i, i \in WikipediaGoodPages} sim(t, w^i).sim(a^j, w^i) \tag{411}$$

As the result of this process, the advertisement with the highest Wikipedia similarity is considered as the most relevant advertisement for the Twitter user. Another important point is that only Equation 41 and Equation 411 are calculated for each user as an online task, other equations are calculated only once as an offline task.

In order to illustrate the similarity calculation, consider a sample with 2 Wikipedia Good Pages $W_1$ and $W_2$, and 3 advertisements $A_1$, $A_2$ and $A_3$. Wikipedia Good Pages $W_1$ *and* $W_2$ contain 3 noun keywords (there may be many words in the body text, but we only consider nouns as keywords). Our method calculates TF-IDF value for each of these keywords as an offline process. The keywords in the pages and their TF-IDF values are as follows:

- Wikipedia Page $W_1$ = {car(0.212), vehicle(0.074), automobile(0.023)}
- Wikipedia Page $W_2$ = {life(0.032), sport(0.0146), accident(0.069)}

Each of advertisements $A_1$, $A_2$, and $A_3$ consists of several keywords, *bid keywords*, to explain the advertisement content. TF-IDF values for advertisements' bid keywords are calculated as an offline process, as in the previous step. The bid keywords in the advertisements and their TF-IDF values are as follows:

- Advertisement $A_1$ (an automobile brand) : {automobile(0.289), sport(0.076), vehicle(0.071)}
- Advertisement $A_2$ (an insurance company): {life(0.083), accident(0.084), insurance(0.037)}
- Advertisement $A_3$ (a game company): {network (0.315), achievement(0.036), game(0.292)}

The proposed method collects the main user's Twitter account profile keywords and calculates top 3 noun keywords with Hybrid TF-IDF values as follows:

- User U = {car(0.247), horsepower(0.0347), accident(0.0526)}

Similarity between User U and advertisements $A_1$, $A_2$, $A_3$ are calculated as follows:

- Sim(U, $A_1$) = Sim(U, $W_1$) . Sim($W_1$, $A_1$) + Sim(U, $W_2$) . Sim($W_2$, $A_1$)
- Sim(U, $A_2$) = Sim(U, $W_1$) . Sim($W_1$, $A_2$) + Sim(U, $W_2$) . Sim($W_2$, $A_2$)
- Sim(U, $A_3$) = Sim(U, $W_1$) . Sim($W_1$, $A_3$) + Sim(U, $W_2$) . Sim($W_2$, $A_3$)

Using cosine similarity on TF-IDF weighted keyword vectors, as given in Equation 410, similarity values are calculated as follows:

- Sim(U, $W_1$) = 0.052364 / (0.254911 * 0.225718) = 0.910076
- Sim(U, $W_2$) = 0.003629 / (0.254911 * 0.077447) = 0.183840
- Sim($W_1$, $A_1$) = (0.00525 + 0.006647) / (0.225718 * 0.307144) = 0.1716
- Sim($W_1$, $A_2$) = 0
- Sim($W_1$, $A_3$) = 0
- Sim($W_2$, $A_1$) = 0.001109 / (0.077447 * 0.307144) = 0.046621
- Sim($W_2$, $A_2$) = 0.008452 / (0.077447 * 0.123749) = 0.881887
- Sim($W_2$, $A_3$) = 0
- Sim(U, $A_1$) = (0.910076 * 0.1716) + (0.18384 * 0.046621) = **0.1647**
- Sim(U, $A_2$) = (0.910076 * 0) + (0.18384 * 0.881887) = **0.162126**
- Sim(U, $A_3$) = (0.910076 * 0) + (0.18384 * 0) = **0**

Results show that advertisement $A_1$ is the most relevant one for user U. One of the most important point in this experiment that, there is no common keywords between advertisement $A_1$ and user U. Wikipedia page usage as the reference pages, overcomes the synonym problem. Advertisement $A_2$ is less relevant and advertisement $A_3$ is irrelevant for the selected user.

# CHAPTER 5

# DIVERSIFICATION

In this chapter, we will explain the third part of the proposed solution: Diversification. Firstly, we will explain previous studies about diversification and after that, we will explain the details of the proposed solution diversification sub-part.

## 5.1 Introduction

Growing industries produce a high number of consumable products and contents. For users, it is very hard to select best goods and for service suppliers, it is very hard to recommend goods to increase user satisfaction. In order to solve this challenge, recommendation is a popular research field and there are a vast number of studies in this area since mid-1990s.

On the other hand, over-specialization (i.e., over-fitting) is one of the most important problems in content based recommendation algorithms. Recommending very similar items repeatedly may be a factor for decreasing user satisfaction. Consider a case where user has just bought a laptop and recommender system recommends other brands laptops. In this scenario, recommending another laptop would not be a feasible suggestion as he has already bought a new one.

As a remedy to this problem, diversification is proposed and it has been studied popularly in the recent years. Over-fitting problem occurs when recommender system concentrates on too similar recommendation items [69]. This problem can be seen when the whole list of recommended items are in the same characteristics. As another example, consider a use who gives high ratings to action films in a movie critics

web site, and recommender system suggests the list of unwatched action films. In this scenario, since the recommender system recommends only action films, user can find the recommendation uninteresting. In this scenario, my prefer to be suggested other types of movies, such as comedy. Actually, diversification in recommender systems is an optimization problem. Recommended items should be similar to user interest and result set items should be dissimilar. Hence, *recommendation diversification* aims to recommend dissimilar items which are relevant to user interests [70].

In this chapter, we study the diversification for advertisement recommendation, to the best of our knowledge, which has not been studied in the literature before. More specifically, we focus on advertisement recommendation on microblogs, and we analyze the diversification performance of a new advertisement recommendation algorithm presented in [22]. It generates recommendation for microblogs, in contrast to advertisement recommendation for web pages in the literature. This algorithm differs from previous algorithms, because microblog domain has its own characteristics and this algorithm uses all functionality of microblogs. Moreover, followee/follower influence analysis is used to extract hidden interests. To evaluate performance result of recommendation algorithm, real world advertisement dataset and public Twitter accounts are collected. Three independent human judges are used to evaluate the results. In evaluation part, different diversification metrics are calculated. This metrics are used to investigate diversification performance of the recommendation algorithm with compared to related works' diversification performances.

## 5.2 Related Works about Diversification

With the rising trend of recommendation systems, they are used in every part of our daily life. Human experiments show that diversification usage increases recommendation satisfaction, therefore there is a high number of diversification studies in the literature. In this section, we summarize the studies on diversification metrics and the works on new techniques to increase diversification values.

The work given in [71] is the one of first diversification studies, and the authors define diversity metric as the average dissimilarity between all pairs of the recommended

items. [72] measures diversity with nDCG coefficient, [73] measures diversity with Gini Coefficient and [74] measures diversification with updated version of nDCG in their studies. In [75], authors provide a mechanism such that the users can manage the diversity value.

In [76], authors develop a framework for diversification of recommendations. In their framework, they supply diversification by expanding the sizes and categories of the user-item interaction records. They compare the performance of five different algorithms under the metrics of precision, recall, f-measure, diversity, tag diversity and tag coverage. In [77] and [78], authors find a new evaluation approach for individual diversity. They evaluated both implicit MMR ([79]) and explicit xQuaD ([80]) methods. In [70] and [81], authors develop an explanation algorithm that can be used in a variety of recommender systems. In their system, each item is defined with explanations and diversification is applied by using these explanations.

In [69], authors make a survey about diversity in recommender systems. They select 39 different articles to investigate. They group articles that define the diversity (11 articles), measure the impact of diversity (10 articles) and propose new diversification algorithms (18 articles).

## 5.3 Proposed Solution

The proposed method [22] is composed of three sub-phases: User interest model construction, advertisement model construction and matching these two models to recommend the best advertisement set. The overview of the proposed solution is presented in Figure 5.1. Whole details of this method can be found in [22], in this chapter we adopt and investigate the proposed method in the diversification manner.

The Similarity between each advertisement and each Wikipedia page is calculated in order to obtain advertisement-Wikipedia page similarity matrix. In this work, we use a different and novel normalization process. For each advertisement, Wikipedia pages are sorted according to similarity values. Each advertisement is represented by top-most similar 100 Wikipedia pages. In this manner, we eliminate the advertisement domination factor. Some advertisements' keywords are included in most of

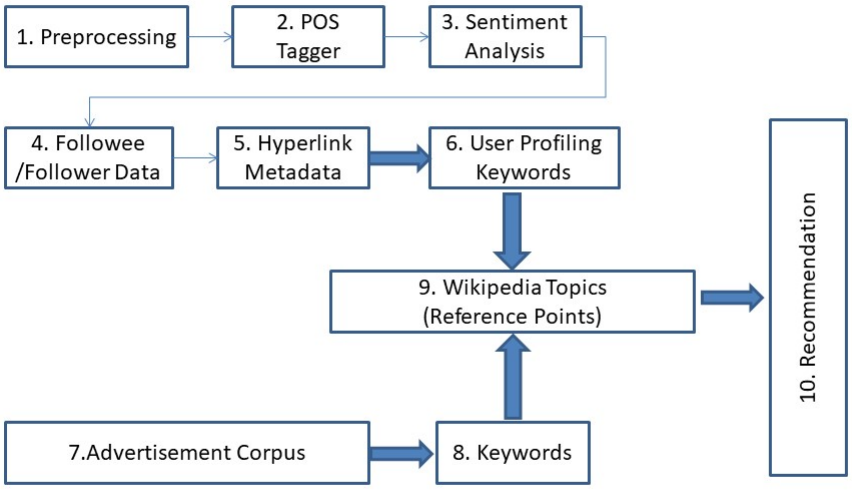



Figure 5.1: Overview of the Proposed Method

the Wikipedia pages. To reduce this boost effect, we represent each advertisement by top-most similar 100 Wikipedia pages. This is the key value of the proposed system for diversification. If the proposed system do not normalize the result, dominating advertisements are always recommended and diversification value will be very small.

As the result of this process, the advertisement with the highest Wikipedia similarity is considered as the most relevant advertisement for the Twitter user.

# CHAPTER 6

# EXPERIMENTS AND ANALYSIS

In this chapter, the proposed solution and its algorithms are evaluated by using performance and accuracy criteria. In evaluation phases, real world datasets are used. Three different and independent human judges are used to compare the proposed solution and the state of the art related work studies.

## 6.1 Dataset

In these experiments, we use 1250 different advertisements as the advertisement corpus. Most of them have been collected from *Fortune Biggest 1000* company list [1]. The company list includes several headings of information per company. Among them, *Industry* field is used for bid-phrases, *Name* field is used as the title of the advertisement, and *Website* field is used as URL of the advertisement. It is a domain independent set including companies from various business areas. Moreover, we additionally constructed 250 up-to-date advertisements manually by querying random keywords in the Google search engine. For each query, top 3 different advertisement brands are selected and corresponding keyword sets are constructed through their web pages. In order to provide diversity, the query keywords are selected randomly by using a random word generator tool.

[1] https://connect.data.com/directory/company/fortune/1000

| Judge | With Sentiment With POS (%) | Without Sentiment With POS(%) | With Sentiment Without POS(%) | Without Sentiment Without POS(%) |
|---|---|---|---|---|
| 1 | 68 | 32 | 0 | 0 |
| 2 | 62 | 38 | 0 | 0 |
| 3 | 64 | 36 | 0 | 0 |
| Avg: | 64,66666667 | 35,33333333 | 0 | 0 |

Figure 6.1: Information extraction algorithms comparison table

## 6.2 User Profiling Experiments and Results

### 6.2.1 Experiment 1: Comparative Analysis of User Profiling

We select Hybrid TF-IDF calculation method of [8] as our baseline in experimental evaluation. We add sentiment feature in order to make better user profiling. Moreover POS tagger is another important property for our evaluation chart, therefore we compare 4 different methods in experiments.

1. With Sentiment With POS Hybrid TF-IDF: Sentiment supported Hybrid TF-IDF algorithm. Our novel approach.

2. Without Sentiment With POS Hybrid TF-IDF: Base algorithm four our study, described in [8].

3. With Sentiment Without POS Hybrid TF-IDF: Sentiment supported non preprocessed algorithm.

4. Without Sentiment Without POS Hybrid TF-IDF: Lack of any intelligent algorithm and preprocessing step.

For the user study, we employed 3 different human judges in order to sort the results of these 4 different versions according to the success to represent a given user. There are 50 different Twitter accounts in our corpus and each Twitter account is represented by 50 different tweets. Figure 6.1 shows the evaluation result. According to this result, Judge 1 selects our solution as the best solution for 34 / 50 of the user accounts. The same judge selects the result by the work in [8] as the best solution for 16 / 50 of the

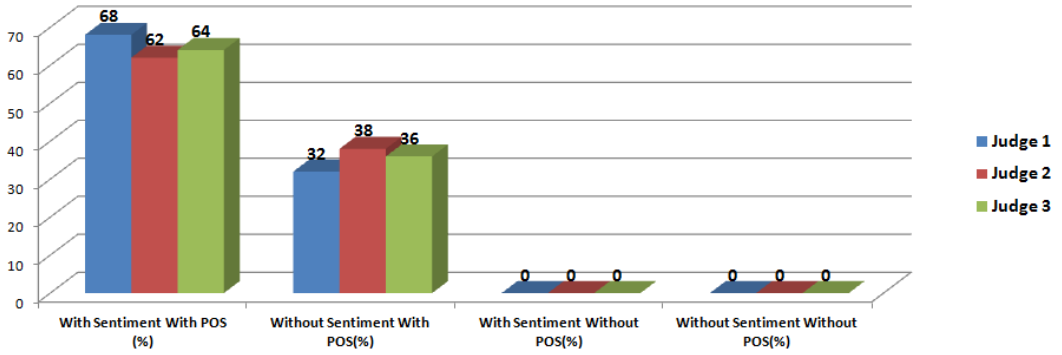



Figure 6.2: Information extraction algorithms comparison chart

user accounts. It shows that, Judge 1 decides Solution 1 is nearly two times better than Solution 2. Like Judge 1, Judge 2 selects our solution for 31 / 50 of the Twitter accounts and Judge 3 selects our solution for 32 / 50 of the accounts. As it can be seen in Figure 6.2 our results are far better than the baseline technique.

### 6.2.2 Experiment 2: User Profiling Precision/Recall Analysis

We evaluate how well the proposed method can extract keywords in comparison to the baseline method in terms of precision and recall. In this experiment, human judge extracts 5 keywords for each of the 50 Twitter users selected for this experiment. This ground truth is compared against the keywords extracted by the proposed and the baseline algorithms.

In this experiment, we can describe our evaluation criteria as follows: For a given user account, if an algorithm matches more number of ground truth keywords, then it is considered to have better result for this account. If both algorithms have the same number of matched keywords, it is a tie situation. According to this criteria, proposed algorithm has better results for 14 Twitter accounts. Baseline algorithm has better result for 6 accounts and they are equal level of success for 30 Twitter accounts. For precision/recall calculation we use 250 keywords for ground truth (5 keywords for each account), and algorithms generate 250 keywords (5 keyword for each account). Table 6.1 shows the precision and recall for the proposed algorithm and the baseline algorithm.

Table 6.1: Precision/Recall

| Algorithm | Precision | Recall |
|---|---|---|
| Proposed Algorithm | 35.6 | 35.6 |
| Baseline study [8] | 30.8 | 30.8 |

One important note about this table is that precision and recall values are the same since the number of retrieved keyword count is equal to number of relevant keywords in formula.

$$precision = \frac{relevant\_keywords \cap retrieved\_keywords}{retrieved\_keywords} \tag{61}$$

$$recall = \frac{relevant\_keywords \cap retrieved\_keywords}{relevant\_keywords} \tag{62}$$

### 6.2.3 Experiment 3: Effect of Different Sentiment Corpora

The basic contribution of the proposed approach is the incorporation of sentiment value into TF-IDF calculation. In our study, we add sentiment feature by using emoticons and subjective words. In order to find words' sentiment values, a look-up table should be used. In the literature, there are several different look-up tables for sentiment values. Therefore, as the last set of experiments, we analyze the effect of using different sentiment corpora on the accuracy of the algorithm. We compared the performance under three most popular sentiment corpora in the literature: Subjectivity Lexicon [43], TwitAtt [4] and Senti-Strength [82]. In Figure 6.3, the results are summarized. As given in the figure, Subjectivity Lexicon gives the best result.

If we compare these sentiment look-up tables; we can easily see that Subjectivity Lexicon is the biggest corpus according to its word count. Moreover, its structure

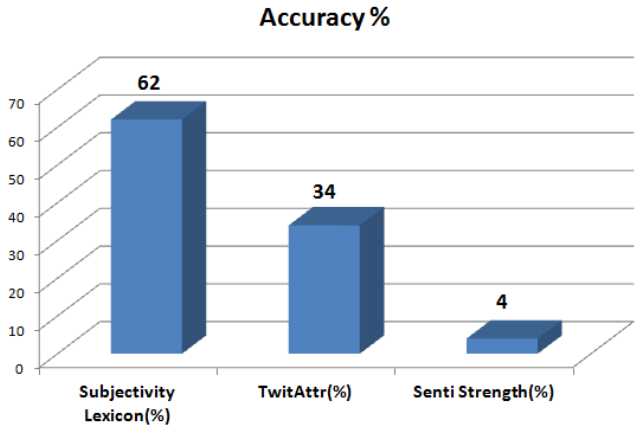



Figure 6.3: Comparison between Sentiment Corpuses

shows that it is a well prepared list. Therefore, this set of sentiment data has the best evaluation result. TwitAtt is a small list but it is specifically prepared for Twitter, most common Twitter words are classified according to sentiment data. It is the second best corpus for this domain. Lastly, Senti-Strength has the least accurate results in our evaluation set.

## 6.3 Advertisement Recommendation Experiments and Results

In this section, the proposed method is analyzed with five different experiments. In the first, second and third experiments, recommendation accuracy is evaluated in comparison to the previous works under different profiling methods. In the fourth experiment, user profiling performance of the proposed model is investigated. In the fifth experiment, the effect of user profile size is evaluated by using recommendation accuracy and run time metrics. Moreover, in the same experiment, the effect of followee/follower enhancement in the user profile on advertisement recommendation accuracy is further analyzed.

### 6.3.1 Experiment 1: Comparative Analysis on the Effect of Including the Followee/Follower Data in the User Profiles

#### 6.3.1.1 Experiment Details

In this experiment, we conduct analysis in comparison to advertisement recommendation technique given in [46]. Randomly selected 200 Twitter accounts are used in the evaluation. Three human judges participated in the study. In each Twitter account, top 3 advertisement recommendations by the technique in [46] and top 3 advertisement recommendations by our method are generated and presented to the judges. We have generated two versions of recommendations: under user profiles with and without followee/follower effect.

In Figure 6.4, we present a textual snapshot of the information presented to the judge for evaluation. An evaluation item consists of the following components:

- The user's latest 100 tweets
- Top 5 extracted keywords (and their TF-IDF values) from user profiling part without followee/follower effect
- Top 5 extracted keywords (and their TF-IDF values) from user profiling part with followee/follower effect
- Top 3 recommended advertisements by using the algorithm in [46] without followee/follower effect
- Top 3 recommended advertisements by using the proposed algorithm without followee/follower effect
- Top 3 recommended advertisements by using the algorithm in [46] with followee/follower effect
- Top 3 recommended advertisements by using the proposed algorithm with followee/follower effect
- User's latest 100 followees and their 100 tweets
- User's latest 100 followers and their 100 tweets

```
---------Result Keywords---------
support TFIDF:0.0656163172300713
learning TFIDF:0.0512033774195705
code TFIDF:0.0427140829401704
github TFIDF:0.041081919929692
branch TFIDF:0.0403473047962028
---After Followee/Flower Data---
support TFIDF:0.0663694433081993
learning TFIDF:0.0515044576241725
code TFIDF:0.0452880323229019
github TFIDF:0.0413403342801515
programming TFIDF:0.0411609680470557
---------Result Advertisements---------
---------Without Followee/Follower---------
Solution 1
Advanced Micro Devices, Inc. : 261.5631
International Business Machines Corporation : 232.7311
Insight Enterprises, Inc. : 199.4959
Solution 2
Adobe Systems Incorporated : 1004.60912435773
Science Applications  International Corporation : 1004.60912435773
Cognizant Technology Solutions Corporation : 1001.90802435773
---------With Followee/Follower---------
Solution 1
Advanced Micro Devices, Inc. : 358.1442
International Business Machines Corporation : 282.522899999999
SunGard : 259.941100000001
Solution 2
Adobe Systems Incorporated : 1032.12912772346
Science Applications  International Corporation : 1032.12912772346
Computer Sciences Corporation : 1030.82922772346
```

Figure 6.4: Evaluation Item

In the evaluation, we use code names for solution by [46] (Solution 1) and for solution by the proposed method (Solution 2) to provide objectivity. Hence, judges know only code names and they have no idea about the algorithms generating the recommendations. In this experiment, for a given user, each judge made a comparison decision for 2 different sets. Comparison between S1 and S2 without Followee/Follower Effect. Another comparison between S1 and S2 with Followee/Follower Effect. 3 Independent Human Judges compare the sets by using the evaluation criteria in Table 6.2.

Table 6.2: Evaluation Criteria

| Solution S1 | Solution S2 | Explanation |
|---|---|---|
| 0 | 0 | Neither of the recommendations are relevant to user interests. |
| 1 | 0 | The first set of recommendations is more relevant to user interests. |
| 0 | 1 | The second set of recommendations is more relevant to user interests. |
| 1 | 1 | Both recommendations are equally relevant to user interests. |

#### 6.3.1.2 Results

Table 6.3 shows the evaluation results for 200 randomly selected Twitter accounts. In this table, *Average Value* means mathematical average value of the judge votes. *Judge Agreement* denotes the majority among 3 judge votes. *Accuracy* denotes the preference percentage over 200 user accounts according to majority voting.

The results show that the proposed method improves recommendation quality in comparison to the solution by [46]. If we use followee/follower effect in user profiling, advertisement quality is further increased. Since the followee/follower effect in user profiling is a contribution of the proposed approach, it would be fair to compare S1 without followee/follower and S2 with followee/follower (76 vs 133), resulting with 75% accuracy increase in advertisement recommendation.

In order to statistically test the consistency between judges, Kappa test is used. The

Table 6.3: Accuracy Results of the Advertisement Recommendation Methods According to Judge Scores

| Method Name | Judge 1 | Judge 2 | Judge 3 | Average Value | Judge Agreement | Accuracy |
|---|---|---|---|---|---|---|
| S1: Method in [46] without Followee/Follower | 75 | 89 | 72 | 78.7 | 76 | 0.38 |
| S2: Proposed Method without Followee/Follower | 127 | 123 | 119 | **123** | **124** | **0.62** |
| S1: Method in [46] with Followee/Follower | 76 | 93 | 77 | 82 | 80 | 0.40 |
| S2: Proposed Method with Followee/Follower | 138 | 126 | 132 | **132** | **133** | **0.67** |

accuracy result between Judge 1 and Judge 2 is "fair" (Kappa score = 0.385). The accuracy result between Judge 1 and Judge 3 is "good" (Kappa score = 0.641). The accuracy result between Judge 2 and Judge 3 is "good" (0.627). These results are calculated with 95% confidence interval.

P-test statistics and null hypothesis testing with 99% confidence interval prove the statistical significance of the results. For each comparison, $Z_0$ value [2] is bigger than 2.33 (99% confidence level for p-test) therefore null hypothesis is rejected, and the difference is statistically significant.

### 6.3.2 Experiment 2: Analysis on the Performance Improvement in Comparison to Keyword Matching Method

In this experiment, we analyze how enrichment strategy improves the recommendation quality in comparison to the Keyword Matching Strategy. In keyword matching, the matching between the keywords of the profile and the advertisements are directly compared.

[2] $Z_0$ = 9.6 (without followee/follower), $Z_0$ = 10.642 (with followee/follower)

#### 6.3.2.1 Experiment Details

Randomly selected 200 Twitter accounts (the same set of users as in Experiment 1) are used in this experiment. In each Twitter account, 3 independent human judges compare the sets by using the evaluation criteria in Table 6.2. An evaluation item consists of the following components:

- The user's latest 100 tweets
- Top 5 extracted keywords (and their TF-IDF values) from user profiling part with followee/follower effect
- Top 3 recommended advertisements by using the Keyword Matching Method with followee/follower effect
- Top 3 recommended advertisements by using the proposed method with followee/follower effect
- User's latest 100 followees and their 100 tweets
- User's latest 100 followers and their 100 tweets

#### 6.3.2.2 Results

Table 6.4 shows the evaluation results for 200 Twitter accounts. In this table, *Average Value* shows arithmetic average value of the judges' votes. *Judge Agreement* denotes the majority among 3 judge votes. *Accuracy* denotes the preference percentage over 200 user accounts according to majority voting.

The results show that the proposed method improves advertisement recommendation quality in comparison to keyword matching method, which is one of the conventional approach for similarity calculation in advertisement recommendation literature.

Table 6.4: Accuracy Results of the Advertisement Recommendation Methods (Keyword Matching Method and Proposed Method)

| Method Name | Judge 1 | Judge 2 | Judge 3 | Average Value | Judge Agreement | Accuracy |
|---|---|---|---|---|---|---|
| S1: Keyword Matching Method with Followee/Follower | 64 | 52 | 68 | 61.33 | 61 | 0.305 |
| S2: Proposed Method with Followee/Follower | 142 | 131 | 143 | **138.66** | **137** | **0.685** |

### 6.3.3 Experiment 3: Analysis on the Performance Improvement in Comparison to Ribeiro-Neto Enrichment Method

In this experiment, we analyze how the proposed enrichment method improves the recommendation quality in comparison to the Ribeiro-Neto [47] enrichment method for advertisement recommendation. Being one of the mostly used semantic matching technique ([45], [50], [51] ...) in the advertisement recommendation literature, Ribeiro-Neto method is selected for comparison. In this experiment, Ribeiro-Neto enrichment method is adapted into microblog domain. In the original algorithm, the main web site is enriched with similar web sites. In this experiment, the main user microblog account is enriched with the followees' and followers' accounts. Except for this adaptation, the original algorithm, which explained in the [47], is implemented.

#### 6.3.3.1 Experiment Details

Randomly selected 200 Twitter accounts (the same set of users as in Experiment 1) are used in this experiment. In each Twitter account, 3 independent human judges compare the sets by using the evaluation criteria in Table 6.2. An evaluation item consists of the following components:

- The user's latest 100 tweets
- Top 5 extracted keywords (and their TF-IDF values) from user profiling part with followee/follower effect

- Top 3 recommended advertisements by using the method in [47]
- Top 3 recommended advertisements by using the proposed method
- User's latest 100 followees and their 100 tweets
- User's latest 100 followers and their 100 tweets

#### 6.3.3.2 Results

Table 6.5 shows the evaluation results for 200 Twitter accounts. In this table, *Average Value* shows the arithmetic average value of the judge votes. *Judge Agreement* denotes the majority among 3 judge votes. *Accuracy* denotes the preference percentage over 200 user accounts according to majority voting.

Table 6.5: Accuracy Results of the Advertisement Recommendation Methods (Riberio-Neto Method and Proposed Method)

| Method Name | Judge 1 | Judge 2 | Judge 3 | Average Value | Judge Agreement | Accuracy |
|---|---|---|---|---|---|---|
| S1: [47] strategy with Followee/Follower | 68 | 62 | 71 | 67 | 69 | 0.345 |
| S2: Proposed Method with Followee/Follower | 138 | 127 | 132 | **132.33** | **131** | **0.655** |

The results show that the proposed enrichment method improves recommendation quality in comparison to the Ribeiro-Neto Enrichment method [47], which is commonly used in advertisement recommendation studies for semantic matching.

### 6.3.4 Experiment 4: Analysis on the Performance of the Constructed User Profile

In this experiment, we analyze how well the followee/followeer inclusion improve user profile.

#### 6.3.4.1 Experiment Details

Randomly selected 200 Twitter accounts (the same set of users as in Experiment 1) are used in this experiment. In each Twitter account, top 5 keywords are extracted and 3 human judges are asked to decide *whether a given keyword in the user profile is related with the user's interest or not*. We conducted two versions of the experiment: user profiling with and without followee/follower effect. For each user, an evaluation item is presented to the each human judge, that contains the following information:

- The user's latest 100 tweets
- Top 5 extracted keywords (and their TF-IDF values) for user profiling without followee/follower effect
- Top 5 extracted keywords (and their TF-IDF values) for user profiling with followee/follower effect
- The user's latest 100 followees and their 100 tweets
- The user's latest 100 followers and their 100 tweets

#### 6.3.4.2 Results

Table 6.6 shows the evaluation results for 200 Twitter accounts. In this table, each $K_i$ column represents scoring for keyword *i*. Keyword 1 is the topmost keyword (with the highest TF-IDF value) that represents user interest. The importance is decreased from 1 to 5. The scores are grouped under two main columns: Keyword accuracy without followee/follower effect and keyword accuracy with followee/follower effect. In this table Judge avg. row represents the average of three judges' votes. Agreement denotes the number relevance under majority voting. Accuracy denotes the relevance of keyword according to average. Results show the fact that followee/follower usage increases the accuracy of user profiling. The importance and the relevance of keywords in the profile increases with the rank. Note that followee/follower inclusion is limited by using $\alpha$ and $\beta$ coefficients. Therefore this inclusion can only change the order of keywords having lower TF-IDF values.

Table 6.6: Keyword Relevance Results According to Judge Scores

| | Without Followee/Follower | | | | | With Followee/Follower | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $K_1$ | $K_2$ | $K_3$ | $K_4$ | $K_5$ | $K_1$ | $K_2$ | $K_3$ | $K_4$ | $K_5$ |
| Judge1 | 158 | 135 | 119 | 101 | 89 | 157 | 143 | 119 | 102 | 105 |
| Judge2 | 128 | 102 | 102 | 90 | 86 | 133 | 103 | 97 | 99 | 83 |
| Judge3 | 151 | 130 | 110 | 99 | 80 | 150 | 130 | 117 | 106 | 98 |
| Avg. | 145.6 | 122.3 | 110.3 | 96.6 | 85 | 146.6 | 125.3 | 111 | 102.3 | 95.3 |
| Aggrement | 147 | 124 | 111 | 95 | 87 | 149 | 126 | 113 | 103 | 97 |
| Accuracy | 0.73 | 0.62 | 0.55 | 0.47 | 0.43 | 0.74 | 0.63 | 0.56 | 0.51 | 0.48 |

According to the results in Table 6.6, average results for the keywords decrease as the rank of the keyword decreases. Hence the keyword rankings are inline with the judge preference. Furthermore, there is an increase in the scores for user profiling with followee/follower effect.

### 6.3.5 Experiment 5: Analysis on the Effect of User Profile Size and Followee / Follower Inclusion

In this experiment, we investigate the effect of profile size on run time and advertisement recommendation accuracy. Furthermore, Wikipedia Good Page coverage and Followee/Follower effect is measured as well.

#### 6.3.5.1 Experiment Details

50 Twitter user accounts that are considered to be *Good Profiled* is used in this experiment. A *Good Profiled Account* is an account such that in Experiment 2 judges gave high score for its keywords. The reason for using this smaller data set is to analyze whether we can further the profile size without comprising accuracy.

For 50 good profiled accounts, we generated advertisement recommendations by the solution in [46], and the proposed method, with four different sizes of user profiles: 2 keywords, 3 keywords, 5 keywords, and 8 keywords. As given in Table 6.7, results show that the proposed algorithm provides about 30% increase in run time efficiency

Table 6.7: Run Time Comparison (Average of 50 accounts is reported in terms of miliseconds)

| # of Profiling Keywords | Without Followee/Follower | | | | With Followee/Follower | | | |
|---|---|---|---|---|---|---|---|---|
| | 2 | 3 | 5 | 8 | 2 | 3 | 5 | 8 |
| S1: Method in [46] (ms) | 1070 | 1273 | 2120 | 3505 | 1296 | 1744 | 2826 | 4436 |
| S2: Proposed Method (ms) | **819** | **958** | **1589** | **2772** | **972** | **1318** | **2220** | **3634** |
| Wikipedia Page # | 3013 | 3984 | 7545 | 11274 | 4143 | 6078 | 9505 | 13115 |

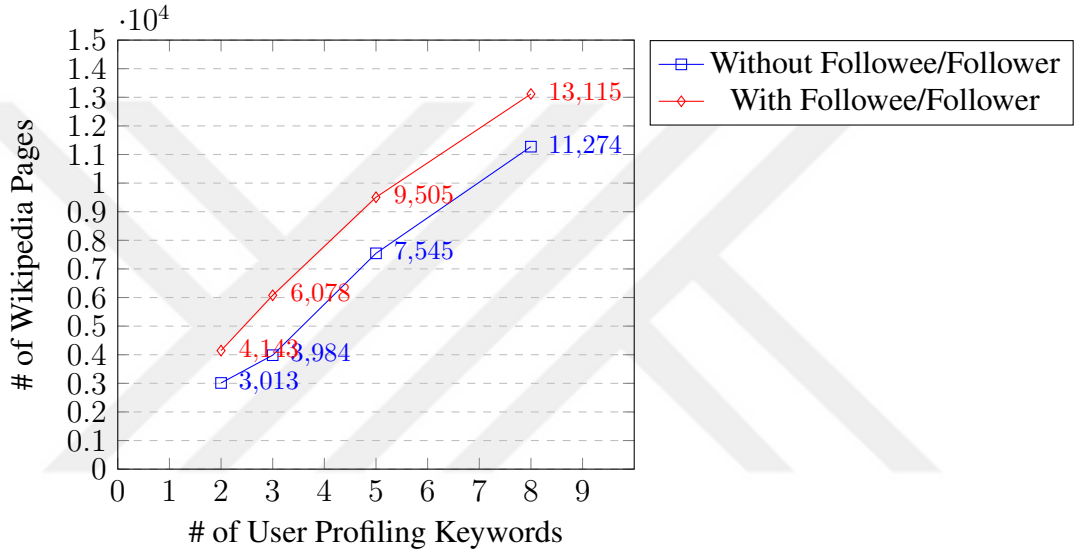



Figure 6.5: The number of Wikipedia Pages covered under the changing size of profiles

in comparison to the solution in [46].

The last row of the table represents how many Wikipedia Good Pages are related with the profile keywords. These coverage values are also plotted in Figure 6.5 in order to show the trend. As expected, 8-keyword profile representation has the highest number of Wikipedia page relations, and the value decreases with the decrease in profile size. We use the same keywords in our algorithm and algorithm in [46] for the sake of fairness. Therefore, related pages counts are same for both solutions. Another important result in this table is that; followee/follower inclusion in user profile increases the number of related Wikipedia page count, and hence increases the coverage of pages.

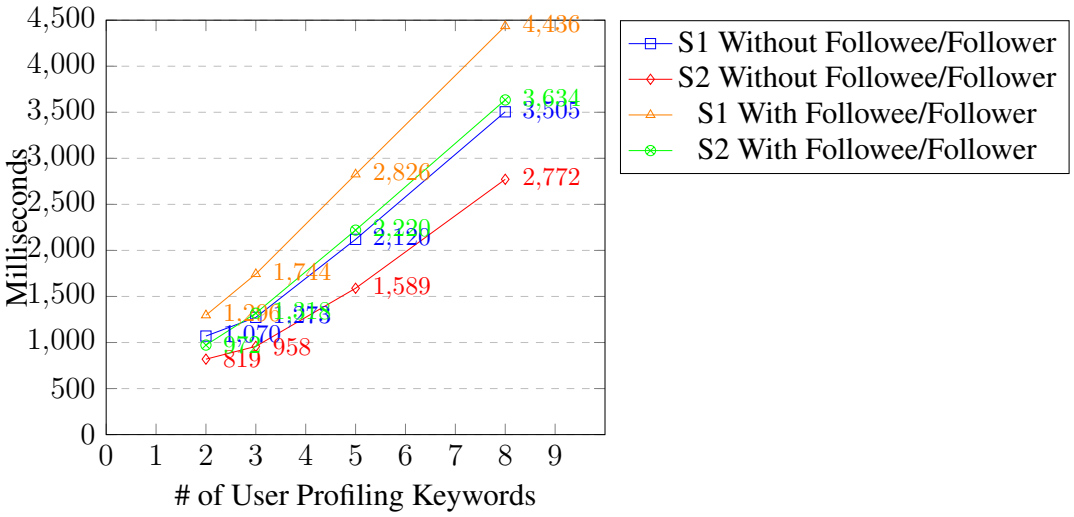



Figure 6.6: Run time in milliseconds under the changing size of profiles

Table 6.8 shows the recommendation accuracy comparison under increasing profile size. As expected, for both algorithms, recommendation accuracy is increased from 2-keyword profiling to 8-keyword profiling. On the other hand, having higher number of keywords in user profile increases the run time values of the algorithms (Figure 6.6).

Moreover, in this experiment, we further investigate the recommendation accuracy to analyze how well the followee/follower inclusion improve the recommendation performance. To investigate the effect of followee/follower, we have generated advertisement recommendations by the proposed solution Without Followee/Follower algorithm and the proposed solution With Followee/Follower algorithm (Table 6.9).

The results given in Table 6.8 is based on the comparison of the judges for the proposed method and the method in [46]. In order to analyze the effect of followee/follower inclusion within the proposed method, we conducted an experiment such that the judges compare the recommendations of two versions of the proposed method (without followee/follower and with followee/follower). Table 6.9 shows that Followee/Follower usage in recommendation supplies about 30% - 40% accuracy improvement. P-test statistics and null hypothesis testing with 99% confidence interval proves the statistical significance of the results. For each keyword profiling level, $Z_0$

Table 6.8: Accuracy Results of the Advertisement Recommendation With Different Profile Size

| # of Profiling Keywords | Method Name | Judge 1 | Judge 2 | Judge 3 | Average Value | Judge Agree | Accuracy |
|---|---|---|---|---|---|---|---|
| 2 | S1: Method in [46] without Followee/Follower | 19 | 20 | 17 | 18.6 | 17 | 0.34 |
| | S2: Proposed Method without Followee/Follower | 27 | 27 | 27 | **27** | **27** | **0.54** |
| | S1: Method in [46] with Followee/Follower | 19 | 19 | 17 | 18.3 | 17 | 0.34 |
| | S2: Proposed Method with Followee/Follower | 34 | 31 | 29 | **31.3** | **33** | **0.66** |
| | S1: Method in [46] without Followee/Follower | 22 | 25 | 20 | 22.3 | 20 | 0.4 |
| 3 | S2: Proposed Method without Followee/Follower | 34 | 31 | 31 | **32** | **32** | **0.64** |
| | S1: Method in [46] with Followee/Follower | 19 | 21 | 20 | 20 | 18 | 0.36 |
| | S2: Proposed Method with Followee/Follower | 36 | 32 | 32 | **33.3** | **34** | **0.68** |
| | S1: Method in [46] without Followee/Follower | 23 | 24 | 22 | 23 | 24 | 0.48 |
| 5 | S2: Proposed Method without Followee/Follower | 41 | 38 | 37 | **38.6** | **39** | **0.78** |
| | S1: Method in [46] with Followee/Follower | 24 | 20 | 22 | 22 | 23 | 0.46 |
| | S2: Proposed Method with Followee/Follower | 41 | 35 | 37 | **37.6** | **41** | **0.82** |
| | S1: Method in [46] without Followee/Follower | 27 | 25 | 22 | 24.6 | 25 | 0.5 |
| 8 | S2: Proposed Method without Followee/Follower | 45 | 37 | 41 | **41** | **42** | **0.84** |
| | S1: Method in [46] with Followee/Follower | 28 | 24 | 25 | 25.6 | 27 | 0.54 |
| | S2: Proposed Method with Followee/Follower | 45 | 41 | 39 | **41.6** | **44** | **0.88** |

Table 6.9: Followee/Follower Effect Analysis on Recommendation Accuracy With Different Profile Size

| # of Profiling Keywords | Method Name | Judge 1 | Judge 2 | Judge 3 | Average Value | Judge Agree | Accuracy |
|---|---|---|---|---|---|---|---|
| 2 | S2: Proposed Method without Followee/Follower | 27 | 26 | 23 | 25.3 | 25 | 0.5 |
| | S2: Proposed Method with Followee/Follower | 34 | 33 | 32 | **33** | **34** | **0.68** |
| 3 | S2: Proposed Method without Followee/Follower | 27 | 28 | 26 | 27 | 23 | 0.46 |
| | S2: Proposed Method with Followee/Follower | 36 | 34 | 38 | **36** | **35** | **0.7** |
| 5 | S2: Proposed Method without Followee/Follower | 34 | 32 | 33 | 33 | 33 | 0.66 |
| | S2: Proposed Method with Followee/Follower | 42 | 37 | 40 | **39.6** | **42** | **0.84** |
| 8 | S2: Proposed Method without Followee/Follower | 36 | 33 | 35 | 34.6 | 35 | 0.7 |
| | S2: Proposed Method with Followee/Follower | 46 | 40 | 45 | **43.6** | **47** | **0.94** |

value [3] is bigger than 2.33 (99% confidence level for p-test) therefore null hypothesis is rejected, and the difference is statistically significant.

## 6.4 Diversification Performance Experiments and Results

In this section, the *Advertisement Recommendation Algorithm* diversification value is compared with base studies diversification values. We want to show that our recommendation algorithm increases recommendation accuracy and recommends more diverse advertisement set. The first part of the sentence "*increases recommendation accuracy*" is proved in the Chapter 4. In this section, we will prove the second part of the sentence ("*recommends more diverse advertisements*").

In this section, we evaluate the performance of advertisement recommendation for

[3] $Z_0$ = 3.66 (2 keyword), $Z_0$ = 4.86 (3 keyword), $Z_0$ = 4.15 (5 keyword), $Z_0$ = 8.85 (8 keyword)

microblogs through diversification perspective. We analyze the diversification performance of the algorithm given in Chapter 4 in comparison to baselines and the state of the art solutions. In Chapter 4, recommendation accuracy and run time efficiency of the algorithm is already presented. Hence, in this section, we focus on diversification evaluation.

### 6.4.1 Dataset

In these experiments, we use 1250 different advertisements as advertisement corpus. Most of them have been collected from Fortune 1000 companies list. We additionally constructed 250 up-to-date advertisements manually by querying random keywords in the Google search engine. For each query, top 3 different advertisement brands are selected and corresponding keyword sets are constructed through their web pages. In order to provide diversity, the query keywords are selected randomly by using a random word generator tool.

### 6.4.2 Setup for all Experiments

For each Twitter account, top 3 advertisements are recommended by each technique. Totally 200 randomly selected Twitter accounts are used for user dataset. Recommendation accuracies are evaluated from human judges and diversification values are calculated automatically by using diversification metrics.

### 6.4.3 Experiment 1: Diversification Value Comparison against Keyword Matching Based Recommendation

#### 6.4.3.1 Experiment Details

In this experiment, we compare the diversification value of general advertisement recommendation technique (keyword matching algorithm) and the recommendation algorithm in [22]. For each Twitter account, top 3 advertisements are recommended by each technique.

#### 6.4.3.2 Diversification Metric

In this experiment, for each technique 3 advertisements are recommended. Each advertisement contains bid keywords. These advertisements and their bid keywords can be represented as sets, therefore we can use Jaccard Index (Similarity Coefficient) and its complement Jaccard Distance as the diversity metric.

Jaccard Coefficient is defined as the number of elements in the intersection of recommendation item sets to the union of them, as given in Equation 63.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{63}$$

Jaccard Coefficient gives the similarity result between two sets, complement of this equation gives the Jaccard Dissimilarity, as given in Equation 64.

$$J_{Dis} = (1 - J(A,B)) * 100 \tag{64}$$

In our experiment, three different advertisements are recommended and we can calculate Jaccard Dissimilarity by getting average of all two distinct set combinations of three advertisements, as given in Equation 65.

$$\overline{J_{Dis}} = \frac{J_{Dis}(1,2) + J_{Dis}(1,3) + J_{Dis}(2,3)}{3} \tag{65}$$

#### 6.4.3.3 Results

Table 6.10 shows the evaluation results for 200 randomly selected Twitter accounts. In this table, *Accuracy* denotes the preference percentage over 200 user accounts according to majority voting of three different human judges. *Diversification* denotes the diversification value of recommended advertisements according to Jaccard Distance value as explained in Section 6.4.3.2.

Table 6.10: Accuracy and Diversification Result

| Method Name | Accuracy | Diversification |
|---|---|---|
| S1: Keyword Matching with Followee/Follower | 0.305 | 74.95 |
| S2: Recommendation Algorithm with Followee/Follower [22] | 0.685 | 80.29 |

The results show that the recommendation algorithm of Chapter 4 has higher recommendation quality and recommends more diverse advertisement list in comparison to keyword matching based recommendation algorithm.

### 6.4.4 Experiment 2: Diversification Value Comparison against the Previous Work

#### 6.4.4.1 Experiment Details

In this experiment, we compare the diversification value of the algorithm in [46] and the recommendation algorithm of [22]. In this experiment, the algorithm in [46] is selected for comparison due to its high accuracy performance in the literature among Wikipedia enrichment based solutions. For each Twitter account, top 3 advertisements are recommended by each technique.

#### 6.4.4.2 Diversification Metric

Jaccard Coefficient and Jaccard Distance Percentage is used as diversification metric (the same metric with Experiment 1)

#### 6.4.4.3 Results

Table 6.11 shows the evaluation results for 200 randomly selected Twitter accounts. In this table, *Accuracy* denotes the preference percentage over 200 user accounts

according to majority voting of three different human judges [22]. *Diversification* denotes the diversification value of recommended advertisement according to Jaccard Distance value as explained in Section 6.4.3.2.

Table 6.11: Accuracy and Diversification Result

| Method Name | Accuracy | Diversification |
|---|---|---|
| S1: Algorithm in [46] with Followee/Follower | 0.40 | 75.23 |
| S2: Recommendation Algorithm with Followee/Follower | 0.67 | 80.29 |

The results show that the advertisement recommendation algorithm in Chapter 4 has higher accuracy and can generate more diverse recommendation list in comparison to the advertisement recommendation of [46] under Jaccard Dissimilarty.

### 6.4.5 Experiment 3: Set Operation of Jaccard Value Analysis in Comparison to Previous Work

#### 6.4.5.1 Experiment Details

In this experiment, we compare the diversification performance of the keyword matching based algorithm, recommendation algorithm in [46] and the recommendation algorithm in Chapter 4 under two different scenarios: with Twitter Followee/Follower and without Twitter Followee/Follower property in user profiling. For each Twitter account, top 3 advertisements are recommended by each technique. In this experiment, diversification value is calculated by using count of recommended distinct advertisements.

#### 6.4.5.2 Diversification Metric

In this experiment, we consider the recommendations generated for each user as a sequence and Jaccard Coefficient compares recommendation items generated for a user

with the set of recommendation items generated for all previous users so far. Consider a sequence of users $U_1$, $U_2$ and $U_3$, and their recommendation items $U_1$, $U_2$, and $U_3$, respectively. Then the sets to be compared for each of users are as follows:
For $U_1$: A = $U_1$, and B = $U_1$
For $U_2$: A = $U_2$, and B = $U_1 \cup U_2$
For $U_3$: A = $U_3$, and B = $U_1 \cup U_2 \cup U_3$

As in previous experiments, Jaccard Distance is used, and Jaccard Coefficient is calculated for A and B ( $J(A, B)$ ). Average Jaccard Dissimilarity gives the average value of all 200 users Jaccard Dissimilarity value, as given in Equation 66.

$$\overline{J_{Dis}} = \frac{\sum_{i=1}^{200} J_{Dis}(i)}{200} \tag{66}$$

#### 6.4.5.3 Results

Table 6.12 shows the evaluation results for 200 randomly selected Twitter accounts. In this table, *Distinct Advertisement* denotes the number of distinct advertisements to recommend 200 Twitter user. *Diversification* denotes the diversification value of recommended advertisement according to Jaccard Distance value as explained in Section 6.4.5.2.

Results given in Table 6.12 show that the recommendation algorithm in Chapter 4 can generate the highest number of distinct advertisement recommendations. Additionally, according to Jaccard Dissimilarity metric, it has most diverse solutions in this comparison.

Table 6.12: Distinct Advertisement and Batch Diversification Results under Jaccard Distance

| Method Name | # of Distinct Advertisement | $J_{Dis}$ Diversification |
|---|---|---|
| S1: Keyword Matching without Followee/Follower | 287 | 95.143 |
| S2: Keyword Matching with Followee/Follower | 287 | 95.142 |
| S3: Algorithm in [46] without Followee/Follower | 178 | 89.295 |
| S4: Algorithm in [46] with Followee/Follower | 183 | 90.67 |
| S5: Recommendation Algorithm without Followee/Follower | 410 | 98.42 |
| S6: Recommendation Algorithm with Followee/Follower | 417 | 98.46 |

### 6.4.6 Experiment 4: Set Operation of Szymkiewicz-Simpson Value Analysis in Comparison to Previous Work

#### 6.4.6.1 Experiment Details

In this experiment, the setting of Experiment 3 is evaluated under Szymkiewicz-Simpson metric.

#### 6.4.6.2 Diversification Metric

In this experiment, we use Szymkiewicz-Simpson coefficient [83], which is calculated as given in Equation 67.

$$SS(A,B) = \frac{|A \cap B|}{min(|A|,|B|)} \tag{67}$$

Szymkiewicz-Simpson coefficient calculates the similarity between two sets as the overlap between the intersection of the sets and the smaller set.

Average Szymkiewicz-Simpson coefficient gives the average value of all 200 users Szymkiewicz-Simpson coefficient value, as given in Equation 68.

$$\overline{SS} = \frac{\sum_{i=1}^{200} SS(i)}{200} \tag{68}$$

#### 6.4.6.3 Results

Table 6.13 shows the evaluation results for 200 randomly selected Twitter accounts. In this table, *Distinct Advertisement* denotes the number of distinct advertisements to recommend 200 Twitter user. *Diversification* denotes the diversification value of recommended advertisement according to Jaccard Distance value as explained in Section 6.4.6.2.

As in the previous experiment, the recommendation algorithm of Chapter 4 generates more diverse recommendations, this time evaluated under Szymkiewicz-Simpson coefficient.

Table 6.13: Distinct Advertisement and Batch Diversification Results under Szymkiewicz-Simpson coefficient

| Method Name | # of Distinct Advertisement | $SS(A, B)$ Diversification |
|---|---|---|
| S1: Keyword Matching without Followee/Follower | 287 | 0.526 |
| S2: Keyword Matching with Followee/Follower | 287 | 0.526 |
| S3: Algorithm in [46] without Followee/Follower | 178 | 0.702 |
| S4: Algorithm in [46] with Followee/Follower | 183 | 0.697 |
| S5: Recommendation Algorithm without Followee/Follower | 410 | 0.319 |
| S6: Recommendation Algorithm with Followee/Follower | 417 | 0.311 |

# CHAPTER 7

# CONCLUSION AND FUTURE WORKS

Microblogs are the newest social networking tools and they are commonly used in the entire world. People can share their opinions, photos, videos and links. People can follow news, events, activities, products, firms, and lots of things. Consequently, people spend lots of time by using this social network applications. To make people life easier, we want to develop a methodology which runs automatically at the background and recommends products according to user preferences. By that way, it causes user to earn time and marketers to gain more money. There are some previous studies at this area but there is no fully functional and efficient system. Some of the previous studies annoys users with questionnaires and some of them uses only a few properties, therefore recommendation results are not complete.

In this thesis, we develop a system and methodology which works in the public account of a Twitter users. By using public data, we extract user preferences and recommend advertisements without annoying the user. We evaluate our system performance with compare to previous studies. According to these results, we enhanced the recommendation capability by comparing previous state-of-the-art studies.

Beside this, according to the literature, recommendation accuracy is increased but it causes a new problem for users: over-fitting. Recommending same types of items can effect user satisfaction. To solve this problem, diversification issue is raised. Diversification in recommendation system has no absolute definition. However, most of the studies use intra-list diversification as the diversification metric. In intra-list diversification, the diversity value is calculated by using the recommended list items.

Like common sense in this issue, the proposed system uses intra-list diversifica-

tion. For each Twitter account, 3 advertisements are recommended and diversification value is calculated between these advertisements. In this study, two different diversification metric is used; Jaccard index and Szymkiewicz-Simpson coefficient (set overlap). These metrics are most common metrics for set similarity and dissimilarity. According to evaluation results, the proposed solution increases diversification value with compare to the generic solution and base solution for Wikipedia enrichment algorithm. Moreover, the proposed solution increases recommendation accuracy too. It means, the proposed solution recommends more accurate and more diverse advertisements with compare to the previous algorithms.

As the future work, recommendation algorithm of this study can be enhanced in the diversification domain. Current algorithm of this study increases the diversification value but recommendation list can be reorganized to increase diversification value. Moreover, followee/follower coefficient analysis can be done to investigate diversification value maximization. Lastly, different normalization methods can be tried to analyze the effect on the recommendation quality.

## REFERENCES


[1] M. Zanker, "Introduction to recommender recommender systems systems," in *Symposium on Applied Computing*, 2010.

[2] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta, "Classifying latent user attributes in twitter," in *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, SMUC 10, (New York, NY, USA), pp. 37–44, ACM, 2010.

[3] W. Wu, B. Zhang, and M. Ostendorf, "Automatic generation of personalized annotation tags for twitter users," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 689–692, Association for Computational Linguistics, 2010.

[4] C. Meador and J. Gluck, "Analyzing the relationship between tweets, box-office performance and stocks," *Methods*, 2009.

[5] Y.-M. Li and Y.-L. Shiu, "A diffusion mechanism for social advertising over microblogs," *Decision Support Systems*, vol. 54, no. 1, pp. 9 – 22, 2012.

[6] I.-H. Ting and C.-S. Yen", "Opinion groups identification in blogosphere based on the techniques of web mining and social networks analysis," in *IACSIT Hong Kong Conferences*, IACSIT 2012, pp. 76–81, 2012.

[7] C. Xu, M. Zhou, F. Chen, and A. Zhou, "Detecting user preference on microblog," in *DASFAA (2)*, pp. 219–227, 2013.

[8] D. Inouye and J. K. Kalita, "Comparing twitter summarization algorithms for multiple post summaries," in *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pp. 298–306, IEEE, 2011.

[9] C. Lu, W. Lam, and Y. Zhang, "Twitter user modeling and tweets recommendation based on wikipedia concept graph," in *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[10] Z. Wu, G. Xu, Y. Zhang, P. Dolog, and C. Lu, "An improved contextual advertising matching approach based on wikipedia knowledge," *Comput. J.*, vol. 55, pp. 277–292, Mar. 2012.

[11] "Number of monthly active twitter users worldwide, https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/," Last access: January 2019.

[12] M. Stelzner, "Social media marketing industry report," 2011.

[13] W. G. Mangold and D. J. Faulds, "Social media: The new hybrid element of the promotion mix," *Business Horizons*, vol. 52, no. 4, pp. 357 – 365, 2009.

[14] T. Dinev, Q. Hu, and A. Yayla, "Is there an on-line advertisers' dilemma? a study of click fraud in the pay-per-click model," *Int. J. Electron. Commerce*, vol. 13, pp. 29–60, Dec. 2008.

[15] S. H. J. L. Wen Hua, Dat T. Huynh and X. Zhou, "Information Extraction From Microblogs: A Survey," *International Journal of Software and Informatics*, 2012.

[16] "Recommendation systems, http://www.gravityrd.com/recommendation-systems," Last access: January 2019.

[17] "Recommender systems, http://en.wikipedia.org/wiki/recommender_system," Last access: January 2019.

[18] J. D. U. Anand Rajaraman, *Mining of Massive Datasets*. Cambridge University Press, 2012.

[19] "Introduction to recommendation systems, http://aimotion.blogspot.com/2009/10/introduction-to-recommendation-systems.html," Last access: January 2019.

[20] "Twitter rank, http://www.alexa.com/siteinfo/twitter.com," Last access: January 2019.

[21] A. Simsek and P. Karagoz, "Sentiment enhanced hybrid tf-idf for microblogs," in *Proceedings of the 2014 IEEE Fourth International Conference on Big Data and Cloud Computing*, BDCLOUD '14, (Washington, DC, USA), pp. 311–317, IEEE Computer Society, 2014.

[22] A. Simsek and P. Karagoz, "Wikipedia enriched advertisement recommendation for microblogs by using sentiment enhanced user profiles," *Journal of Intelligent Information Systems*, pp. 1–25, 2018.

[23] W. Zeng, Y. Huang, and L. Jiang, "The study of microblog marketing based on social network analysis," in *Information Management, Innovation Management and Industrial Engineering (ICIII), 2011 International Conference on*, vol. 3, pp. 410–415, 2011.

[24] I.-H. Ting, P. S. Chang, and S.-L. Wang, "Understanding microblog users for social recommendation based on social networks analysis," vol. 18, pp. 554–576, feb 2012.

[25] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts," in *Proceedings of EMNLP*, vol. 4, Barcelona, Spain, 2004.

[26] E. Cambria, M. Grassi, A. Hussain, and C. Havasi, "Sentic computing for social media marketing," *Multimedia Tools Appl.*, vol. 59, pp. 557–577, July 2012.

[27] "Live journal, http://www.livejournal.com," Last access: January 2019.

[28] Y.-M. Li and T.-Y. Li, "Deriving marketing intelligence over microblogs," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pp. 1–10, 2011.

[29] S. G. Esparza, M. P. O. Mahony, and B. Smyth, "Effective product recommendation using the real-time web," in *SGAI Conf.10*, pp. 5–18, 2010.

[30] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis.," in *IJCAI*, vol. 7, pp. 1606–1611, 2007.

[31] "Twitter developers, https://developer.twitter.com/en/docs," Last access: January 2019.

[32] "Twitter api call limits, https://dev.twitter.com/docs/rate-limiting/1.1/limits," Last access: January 2019.

[33] W. X. Zhao, S. Li, Y. He, E. Y. Chang, J.-R. Wen, and X. Li, "Connecting social media to e-commerce: Cold-start product recommendation using microblogging information," *IEEE Trans. on Knowl. and Data Eng.*, vol. 28, pp. 1147–1159, May 2016.

[34] P. Bicalho, M. Pita, G. Pedrosa, A. Lacerda, and G. L. Pappa, "A general framework to expand short text for topic modeling," *Information Sciences*, vol. 393, no. Supplement C, pp. 66 – 81, 2017.

[35] F. Riquelme and P. González-Cantergiani, "Measuring user influence on twitter: A survey," *Information Processing & Management*, vol. 52, no. 5, pp. 949 – 975, 2016.

[36] Z. Jiantao and S. Ning, "User interest prediction in microblog using recommendation method," in *Information Technology and Artificial Intelligence Conference (ITAIC), 2014 IEEE 7th Joint International*, pp. 367–370, IEEE, 2014.

[37] H. Zhang and G. Zhong, "Improving short text classification by learning vector representations of both words and hidden topics," *Knowledge-Based Systems*, vol. 102, no. Supplement C, pp. 76 – 86, 2016.

[38] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing & Management*, vol. 53, no. 4, pp. 764 – 779, 2017.

[39] H. Saif, Y. He, M. Fernandez, and H. Alani, "Contextual semantics for sentiment analysis of twitter," *Information Processing & Management*, vol. 52, no. 1, pp. 5 – 19, 2016. Emotion and Sentiment in Social and Expressive Media.

[40] C. Hung, "Word of mouth quality classification based on contextual sentiment lexicons," *Information Processing & Management*, vol. 53, no. 4, pp. 751 – 763, 2017.

[41] C. Standing, M. Holzweber, and J. Mattsson, "Exploring emotional expressions in e-word-of-mouth from online communities," *Information Processing & Management*, vol. 52, no. 5, pp. 721 – 732, 2016.

[42] "The stanford natural language processing group, http://nlp.stanford.edu/software/tagger.shtml," Last access: January 2019.

[43] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 347–354, Association for Computational Linguistics, 2005.

[44] "Iab/pwc digital advertising revenue report, http://www.iab.net/media/file/pwc_iab_2014_full_year_digital_ad_revenue _iab_presentation_all_3_webinar....pdfs," Last access: January 2019.

[45] A. Broder, M. Fontoura, V. Josifovski, and L. Riedel, "A semantic approach to contextual advertising," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, (New York, NY, USA), pp. 559–566, ACM, 2007.

[46] A. Pak and C.-W. Chung, "A wikipedia matching approach to contextual advertising," *World Wide Web*, vol. 13, no. 3, pp. 251–274, 2010.

[47] B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. Silva de Moura, "Impedance coupling in content-targeted advertising," in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, (New York, NY, USA), pp. 496–503, ACM, 2005.

[48] P. N. Mendes, M. Jakob, and C. Bizer, "Dbpedia: A multilingual cross-domain knowledge base," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012), Istanbul, Turkey, May 23-25, 2012*, pp. 1813–1817, 2012.

[49] Z. Wu, G. Xu, R. Pan, Y. Zhang, Z. Hu, and J. Lu, "Leveraging wikipedia concept and category information to enhance contextual advertising," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, (New York, NY, USA), pp. 2105–2108, ACM, 2011.

[50] D. Tsatsou, F. Menemenis, I. Kompatsiaris, and P. C. Davis, "A semantic framework for personalized ad recommendation based on advanced textual analysis,"

in *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, (New York, NY, USA), pp. 217–220, ACM, 2009.

[51] W. Zhang, D. Wang, G.-R. Xue, and H. Zha, "Advertising keywords recommendation for short-text web pages using wikipedia," *ACM Trans. Intell. Syst. Technol.*, vol. 3, pp. 36:1–36:25, Feb. 2012.

[52] P. Wang, J. Hu, H.-J. Zeng, and Z. Chen, "Using wikipedia knowledge to improve text classification," *Knowl. Inf. Syst.*, vol. 19, pp. 265–281, May 2009.

[53] S. Vargas and P. Castells, "Exploiting the diversity of user preferences for recommendation," in *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*, OAIR '13, (Paris, France), pp. 129–136, 2013.

[54] X. W. Zhao, Y. Guo, Y. He, H. Jiang, Y. Wu, and X. Li, "We know what you want to buy: A demographic-based system for product recommendation on microblogs," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, (New York, NY, USA), pp. 1935–1944, ACM, 2014.

[55] W. X. Zhao, S. Li, Y. He, L. Wang, J.-R. Wen, and X. Li, "Exploring demographic information in social media for product recommendation," *Knowl. Inf. Syst.*, vol. 49, pp. 61–89, Oct. 2016.

[56] Q. Yuan, G. Cong, K. Zhao, Z. Ma, and A. Sun, "Who, where, when, and what: A nonparametric bayesian approach to context-aware recommendation and search for twitter users," *ACM Trans. Inf. Syst.*, vol. 33, pp. 2:1–2:33, Feb. 2015.

[57] X. Zhou, S. Wu, C. Chen, G. Chen, and S. Ying, "Real-time recommendation for microblogs," *Information Sciences*, vol. 279, no. Supplement C, pp. 301 – 325, 2014.

[58] D. Karatay and P. Senkul, "User interest modeling in twitter with named entity recognition," in *CEUR Workshop Proceedings*, pp. 17–20, 01 2015.

[59] L.-F. Lin, Y.-M. Li, and W.-H. Wu, "A social endorsing mechanism for target advertisement diffusion," *Information & Management*, vol. 52, no. 8, pp. 982 – 997, 2015.

[60] Y. Jiang, W. Bai, X. Zhang, and J. Hu, "Wikipedia-based information content and semantic similarity computation," *Information Processing & Management*, vol. 53, no. 1, pp. 248 – 265, 2017.

[61] Y. Jiang, X. Zhang, Y. Tang, and R. Nie, "Feature-based approaches to semantic similarity assessment of concepts using wikipedia," *Information Processing & Management*, vol. 51, no. 3, pp. 215 – 234, 2015.

[62] Z. Wu, H. Zhu, G. Li, Z. Cui, H. Huang, J. Li, E. Chen, and G. Xu, "An efficient wikipedia semantic matching approach to text document classification," *Information Sciences*, vol. 393, no. Supplement C, pp. 15 – 28, 2017.

[63] M. A. M. García, R. P. Rodríguez, and L. A. Rifón, "Wikipedia-based cross-language text classification," *Information Sciences*, vol. 406, no. Supplement C, pp. 12 – 28, 2017.

[64] D.-T. Vo and C.-Y. Ock, "Learning to classify short text from scientific documents using topic models with various types of knowledge," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1684 – 1698, 2015.

[65] M. Mehdi, C. Okoli, M. Mesgari, F. Årup Nielsen, and A. Lanamäki, "Excavating the mother lode of human-generated text: A systematic review of research that uses the wikipedia corpus," *Information Processing & Management*, vol. 53, no. 2, pp. 505 – 529, 2017.

[66] G. Xu, Z. Wu, G. Li, and E. Chen, "Improving contextual advertising matching by using wikipedia thesaurus knowledge," *Knowl. Inf. Syst.*, vol. 43, pp. 599–631, June 2015.

[67] "Wikipedia:good articles, https://en.wikipedia.org/wiki/wikipedia:good_articles," Last access: January 2019.

[68] "Wikipedia:good article criteria, https://en.wikipedia.org/wiki/wikipedia:good_article_criteria," Last access: January 2019.

[69] M. Kunaver and T. Požrl, "Diversity in recommender systems – a survey," *Knowledge-Based Systems*, vol. 123, pp. 154 – 162, 2017.

[70] C. Yu, L. Lakshmanan, and S. Amer-Yahia, "It takes variety to make a world: Diversification in recommender systems," in *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '09, (New York, NY, USA), pp. 368–378, ACM, 2009.

[71] K. Bradley and B. Smyth, "Improving recommendation diversity," in *Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland*, pp. 85–94, Citeseer, 2001.

[72] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon, "Novelty and diversity in information retrieval evaluation," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, (New York, NY, USA), pp. 659–666, ACM, 2008.

[73] D. M. Fleder and K. Hosanagar, "Recommender systems and their impact on sales diversity," in *Proceedings of the 8th ACM Conference on Electronic Commerce*, EC '07, (New York, NY, USA), pp. 192–199, ACM, 2007.

[74] S. Vargas, "New approaches to diversity and novelty in recommender systems," in *Proceedings of the Fourth BCS-IRSG Conference on Future Directions in Information Access*, FDIA'11, (Swindon, UK), pp. 8–13, BCS Learning & Development Ltd., 2011.

[75] T. Aytekin and M. O. Karakaya, "Clustering-based diversity improvement in top-n recommendation," *J. Intell. Inf. Syst.*, vol. 42, pp. 1–18, Feb. 2014.

[76] Z. Zhang, X. Zheng, and D. D. Zeng, "A framework for diversifying recommendation lists by user interest expansion," *Knowledge-Based Systems*, vol. 105, pp. 83 – 95, 2016.

[77] T. Di Noia, V. C. Ostuni, J. Rosati, P. Tomeo, and E. Di Sciascio, "An analysis of users' propensity toward diversity in recommendations," in *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, (New York, NY, USA), pp. 285–288, ACM, 2014.

[78] T. D. Noia, J. Rosati, P. Tomeo, and E. D. Sciascio, "Adaptive multi-attribute di-

versity for recommender systems," *Information Sciences*, vol. 382-383, pp. 234 – 253, 2017.

[79] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, (New York, NY, USA), pp. 335–336, ACM, 1998.

[80] R. L. Santos, C. Macdonald, and I. Ounis, "Exploiting query reformulations for web search result diversification," in *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, (New York, NY, USA), pp. 881–890, ACM, 2010.

[81] C. Yu, L. V. S. Lakshmanan, and S. Amer-Yahia, "Recommendation diversification using explanations," *2009 IEEE 25th International Conference on Data Engineering*, pp. 1299–1302, 2009.

[82] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, pp. 2544–2558, 2010.

[83] V. M K and K. K, "A survey on similarity measures in text mining," *Machine Learning and Applications: An International Journal*, vol. 3, pp. 19–28, 03 2016.

# APPENDIX A

## WIKIPEDIA SAMPLES


Figure A.1: Wikipedia Good Pages' Nouns


Figure A.2: Wikipedia Good Pages' Nouns TFIDF values

# APPENDIX B

## ADVERTISEMENT SAMPLES


Figure B.1: Advertisements' Keywords

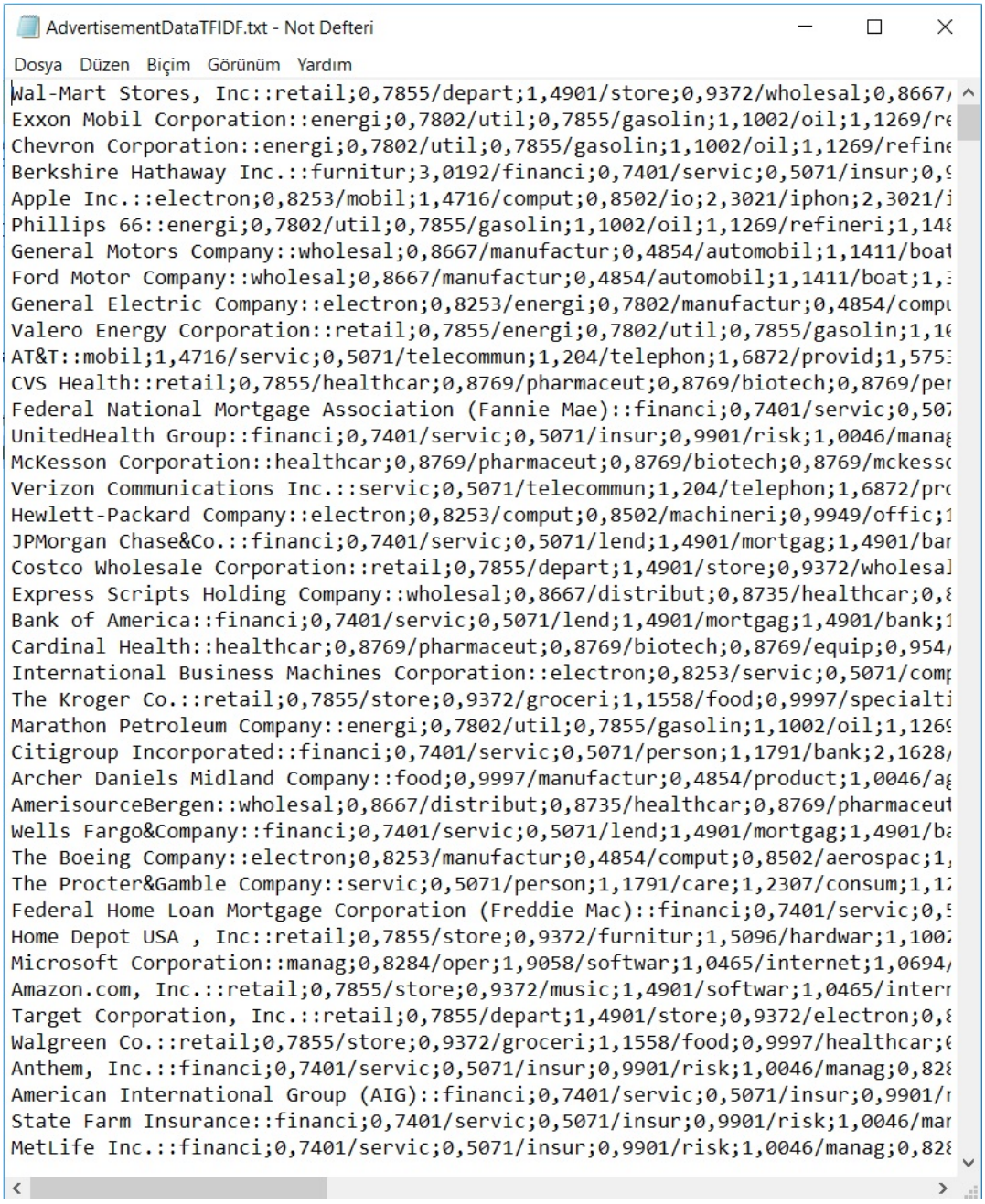



Figure B.2: Advertisements' Keywords TF-IDF values


Figure B.3: Advertisement and Wikipedia Good Page Similarity

## CURRICULUM VITAE

### PERSONAL INFORMATION


| | | |
|---|---|---|
| Surname, Name | : | Şimşek, Atakan |
| Nationality | : | Turkish (TC) |
| Date and Place of Birth | : | 6 September 1983, Malatya |
| Phone | : | +90 506 340 22 44 |
| E-mail | : | e129830@metu.edu.tr |


### EDUCATION

| Degree | Institution | Year of Graduation |
|---|---|---|
| MS | METU Computer Engineering | 2009 |
| BS | METU Computer Engineering | 2006 |
| High School | Malatya Science High School | 2001 |

### WORK EXPERIENCE

| Year | Place | Enrollment |
|---|---|---|
| 2018- Present | VLMedia | Head of Engineering |
| 2006-2018 | TUBİTAK BİLGEM | Chief Engineer |

### FOREIGN LANGUAGES

Advanced English

### PUBLICATIONS


1. A. Simsek and P. Karagoz, "Wikipedia enriched advertisement recommendation for microblogs by using sentiment enhanced user profiles," Journal of Intelligent Information Systems, pp. 1–25, 2018. https://doi.org/10.1007/s10844-018-0540-5

2. A. Simsek and P. Karagoz, "Sentiment enhanced hybrid tf-idf for microblogs," in Proceedings of the 2014 IEEE Fourth International Conference on Big Data and Cloud Computing, BDCLOUD ’14, (Washington, DC, USA), pp. 311–317, IEEE Computer Society, 2014.

3. Simsek, Atakan et al. "KABAN-2: Kullanıcı AraBirimi ve Geçerleme Altyapısı." UYMS, 2014.

**HOBBIES**

Tennis, Swimming, Games