Gazi Mühendislik Bilimleri Dergisi
2018, 4(3): 183-195
Araştırma Makalesi/Research Article
gmbd.gazipublishing.com

Twitter Üzerindeki Etkili Bireylerin Makine Öğrenmesi Sınıflandırma Algoritmaları İle Tespiti

Mehmet ŞİMŞEK[*,a], Abdullah Talha KABAKUŞ[b]

[a,*] *Düzce Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü, DÜZCE 81620, TÜRKİYE*
[b] *Düzce Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü, DÜZCE 81620, TÜRKİYE*

MAKALE BİLGİSİ

Alınma: 08.10.2018
Kabul: 06.11.2018

Anahtar Kelimeler:
Etki Maksimizasyonu,
Twitter,
Sosyal Ağlar,
Mikroblog,
Sınıflandırma

[*]*Sorumlu Yazar:*
e-posta:
mehmetsimsek@duzce.edu.tr

ÖZET

Mikroblog siteleri insanların birbirlerini takip ettikleri ortamlardır. Bu özellikleri ile bir microblog sitesi bir fikrin ya da yeni bir ürünün yayılması için elverişli bir ortamdır. Buradaki anahtar nokta, yayılımı maksimize edecek bireylerin tespitidir. Bu problem, Etki Maksimizasyonu (EM) olarak bilinir ve birçok araştırmacının ilgisini çekmiştir. Literatürdeki birçok çalışma EM problemini graflar üzerinde Independent Cascade (IC) ve Linear Threshold (LT) yayılım modelleri için ele almıştır. Ne var ki, Twitter gibi microblog sitelerinin kendi özellikleri ve vardır. Twitter üzerinde EM problemini ele almış olan birçok çalışma, kullanıcı ve tweet özelliklerinden yeni ölçütler geliştirme ve bu ölçütleri kullanan bir açgözlü algoritma ile etkin bireyleri seçme yolunu izler. Bu çalışmada biz EM problemi farklı bir yaklaşım uyguladık ve problemi bir sınıflandırma problemi olarak ele aldık. İlk olarak, 2018 Uluslararası Kadınlar Gününde veri topladık; kullanıcıları deneysel olarak etkili bireyler ve etkili olmayan bireyler olarak etiketledik; son olarak bireyleri etkili ya da etkili olmayan diye sınıflara ayırmak için sınıflandırma algoritmalarını kullandık. Bu şekilde, ana verisetinden oldukça küçük olan bir etkili bireyler kümesi elde ettik. Deneysel sonuçlar, aynı parametreyi kullanarak indirgenmiş kümeden seçim yapılmasının, bütün veriseti üzerinden seçim yapılmasına göre çok daha başarılı sonuçlar verdiğini göstermiştir.
DOI: 10.30855/GJES.2018.04.03.005

Finding Influencers on Twitter with Using Machine Learning Classification Algorithms

ARTICLE INFO

Received: 08.10.2018
Accepted: 06.11.2018

Keywords:
Influence Maximization,
Twitter,
Social Networks,
Microblog,
Classification

[*]***Corresponding Authors***
e-mail:
mehmetsimsek@duzce.edu.tr

ABSTRACT

Microblog sites are environments where people follow people. With this feature, a microblog site is a convenient environment for spreading an opinion or introducing a new product. The key point is determination of individuals who maximize the spreading. This problem is known as Influence Maximization (IM) and has attracted attention of many researchers. Many studies in the literature have modeled IM problem on graphs for different propagation models such as Independent Cascade (IC) and Linear Threshold (LT). However, microblogs like Twitter have their own features. Many works on IM in Twitter derive new metrics from user and tweet features; apply a greedy approach for selecting influencers. In this study, we adopted different approach for IM problem, and we dealt it as a classification problem. Firstly, we collected data on International Women Day 2018; empirically we labeled the users as either influencer candidates or non-influencers; then we applied classification methods for classifying users into one class with using features of users. By this way, we obtained an influencer candidates set, which is very smaller than entire dataset. Experimental results show that making selection with using same heuristic (namely MF) from the reduced influencer candidates set outperforms making selection from entire dataset.
DOI: 10.30855/GJES.2018.04.03.005

1. INTRODUCTION *(GİRİŞ)*

Social media has become a great media where people share their feelings and thoughts and are influenced by each other. Therefore, it is very suitable for many fields such as shaping public opinion and viral marketing [1]–[10]. Such practices are based on the adoption of an opinion (or a product) by as many people as possible. In the literature, the name of this problem is Influence Maximization (IM). In IM problem, the key point is the determination of who will spread the influence. In this context, the IM problem aims to identify the minimum number of seed individuals that will maximize the spread of an effect [4]. Many studies in the literature have modeled the IM problem on graphs for different propagation models such as Independent Cascade (IC) and Linear Threshold (LT). Kempe et al. suggest a greedy algorithm that uses an approach to select seed nodes one-by-one [4]. By contrast, Borgatti et. al., considered the IM problem as a combinatorial optimization problem and stated that the selection of seeds should be done at simultaneously [11]. Most of the later studies have tried to solve the IM problem either by developing a greedy algorithm or by using an optimization method [5], [12]–[19]. The common feature of these studies is that they address the IM problem on a graph-based basis and try to solve the IM problem by using different centrality metrics (or suggesting new ones). Some of the centrality metrics are based on nodal features, and some on edge features. However, microblogs like Twitter have their own characteristics, and these characteristics provide important information about the individual's activity (influence level) [20], [21]. Therefore, these features are widely used in IM studies for a specific social networking platform. In this study, we dealt with IM problem on Twitter. As we will discuss in the Related Work section, many works adopt the following approach: deriving new metrics from the user and tweet features; applying a greedy approach of sorting individuals according to these metrics and taking the top-k individual as the most influential individuals. We adopted a little bit different approach for IM problem, and we dealt it as a classification problem. Our approach is to classify individuals as influencer candidates and non-influencers, using multiple features, rather than developing a new metric. For this purpose, we have collected tweets on International Women's Day 2018. We calculated Spread Scores [21] for 168.168 unique users. Spread Score is a novel score that indicates a user's potential to spread an information. Empirically, we labeled the users who have Spread Score greater than mean+2×standart_deviation as influencer candidates (class 1), and we labeled the others as non-influencers (class 0). We carried out several experiments for different classification methods with using Weka [22]. Inputs are the features that gathered from Twitter, and the nominal value is the classes (class 1 and class 0) for the classifiers. As a result of the classification, we were able to classify more than half of the most influential users correctly. When we analyze the results, we have seen that the most influential users belong to the person (not the institutions or organizations). As a result, we offered a new heuristic, and we compared our recommendation with the most followed user (MF) heuristic.

The contributions of this paper are summarized as follows:

- By evaluating many features together, we have proposed and successfully implemented the classification of individuals (i.e. Twitter users) with using machine learning methods. As a result, we significantly reduced the problem set.
- Selection of top-k influencers from the reduced influencer candidates set with using same heuristic outperforms the selection from entire dataset.
- We have seen that natural persons are more effective than organizations in situations that directly concern individuals (i.e. congratulations).
- To the best of our knowledge, this work is the first study that investigates IM on a casual topic instead of generic topics (i.e. love, soccer, fun, music, etc.) as the related work handles the IM on Twitter.
- Automated services are implemented in order to fetch and store the real-time tweets in the database, and update the required fields of stored tweets. This is a key necessity when the huge number of tweets is considered.

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 describes the material and method. The experimental result is presented in Section 4. Section 5 describes the discussion. Finally, Section 6 concludes the paper with future directions.

2. RELATED WORK *(İLGİLİ ÇALIŞMALAR)*

We organized this section as two sub-sections. In the first part, we talked about graph-based approaches that deal with the IM problem. In the second part, we gave IM studies on Twitter which is the subject of this study.

2.1. Influence Maximization *(Etki Maksimizasyonu)*

As we mentioned in the introduction, many studies in the literature have often dealt with the IM problem using different propagation models on graphs. In doing so, they used only centrality metrics related to graphs (nodal or edge). Although these metrics provide valuable information about an individual's position on the network, they do not provide information about the individual or the subject matter.

However, there is a need much more information to address the IM problem in a real social networking platform. For example, a person may not be effective on a topic while he is effective at another topic. Similarly, a person may not be effective on a language while he is effective on another language. With using only graph-based metrics, it is not possible to predict the effect that will be created by a person [1], [23], [24]. For this reason, many of the new studies have focused on the use of platform-specific features [20], [21]. These features include the number of follower users, the number of following users, verification status, number of lists the user has added etc. Also, there are many features related to tweets such as the number of retweets, number of likes etc. We will give the all of these features under the Material and Method section. Whether it is graph-based or platform-specific, it is necessary to determine the influence levels of the individuals for evaluating proposed IM method [25]. Graph-based methods try to predict with using IC or LT propagation models that how many individuals will be influenced. Generally, on Twitter, the impacts of persons are measured by the number of retweets that have been received by their tweets [21], [26]. Twitter API does not give information about who retweets whom. Because of this, we cannot find out who spreads the information on Twitter. This makes it impossible to use spreading models such as IC and LT. For this reason, it seems to be the most suitable way is the use of platform-specific features to identify individuals who maximize influence on platforms such as Twitter.

2.2. Twitter Influencer Detection *(Twitter'da Etkili Birey Tespiti)*

Zengin Alp and Gunduz Oguducu [21] present an approach called as "Personalized PageRank" which integrates both the information obtained from the Twitter network and the user activities. The proposed approach aims to determine topic-specific influencers who are believed to be the experts on the selected topic. The features they used to model the proposed approach are (1) the number of tweets of the user, (2) the number of tweets by the user on the selected topic, (3) the number the retweeted tweets of the user, (4) the number the retweeted tweets of the user on the selected topic, (5) the number of retweets of the user, the number of retweets of the user on the selected topic, (6) the total number of days, (7) the number of days the user posted on the selected topic, and (8) the duration passed for the first retweet of the post. Cataldi and Aufaure [25] propose an approach that analyzes the multiple paths the information follows over the network and provides a method for estimating the influence among users by evaluating the relationship among them. They model the Twitter network for a selected topic using a directed graph where the directed edges represent a retweet action between the nodes (users). They also consider authorities of users while calculating influences. Kwak et al. [27] compare different influence measurements based on parameters such as the number of retweets, the number of followers. They also propose "retweet trees" but they do not include this feature as a part of influence measurement. Cha et al. [1] propose an in-depth comparison of three measures of influence namely in-degree, retweets, and mentions. Based on these three measurements, they investigate the influence across time and topics. They report the finding that the most followed users are not necessarily influential in terms of gained retweets or mentions. Liu et al. [6] propose a generative graphical model that utilizes both heterogeneous link information and textual content associated with each user in the network to measure topic-level influence. They apply the proposed approach in four different genres of social networks including Twitter in order to reveal its effectiveness. TwitterRank [28] is an extension of PageRank algorithm proposed to measure the influence of Twitter users. It uses a directed graph D(V, E) which V (Vertex) represents the Twitter users, and E (Edge) represents the following relationships among them. E is directed from follower to friend. Web Ecology project [29] measures the influence based on the ratio of the attention (namely retweet, reply, and mention) the user receives to the total number of tweets the user has.

As a result, the great majority of the studies conducted in the literature are subject-based and require long-term observation. However, an important feature of Twitter is that it is a platform where day-to-day issues [30]–[32] are discussed in a short period of time. With this point of view, we have concentrated on the problem of identifying influential individuals on a day-to-day topic.

3. MATERIALS AND METHODS *(Materyal ve Yöntemler)*

In this section, the details of (1) how the data is collected, (2) how the collected tweets are updated, (3) the features used for machine learning algorithms, (4) the labeling process, and (5) compared classification methods are described in the following subsections.

3.1. Data Collection *(Veri Toplama)*

Twitter provides enhanced APIs (Application Programming Interface) in order to collect, query, and update its data programmatically. Since we need a daily topic for this study, we have decided to fetch the tweets related to "International Women's Day" which is a global event and celebrated on March 8 every year. On this day, a service implemented based on Java programming language which utilizes twitter4j[1], an open source Java library for Twitter APIs, is executed in order to collect tweets through the Twitter Streaming API. Three keywords which are directly related to "International Women's Day" and amongst to the trending topics about this day namely "InternationalWomensDay", "IWD2018", and "WomensDay" are used to filter streaming tweets. The language of tweets which are fetched through the Twitter Streaming API is set to English. With considering the time differences amongst on various continents, the developed service has run for 48 hours. During this time, 219,076 tweets have collected from 168.168 unique Twitter users (for dataset: http://bigdata.duzce.edu.tr/#datasets). The features extracted from the tweets and its owners (Twitter users) using Twitter API are listed in Table 1 & 2, respectively.

The process of data collection is presented in Fig. 1 as (1) presents the process of querying tweets through the provided hashtags, (2) presents the process of returning tweets and their owners according to the provided hashtags, and (3) presents the process of storing the fetched tweets in MongoDB.

Table 1. The features extracted from the collected tweets through the Twitter API *(Twitter API ile toplanan tweetlerden elde edilen özellikler)*

Feature	Description
tweetId	The id of the tweet
userId	The id of the tweet's owner
username	The username of the tweet's owner
message	The content of the tweet
favoriteCount	The number of likes the tweet is gained
rtCount	The number of retweets the tweet is gained
latitude	The latitude information of the tweet (if it is available)
longitude	The longitude information of the tweet (if it is available)
lang	The language of the tweet
mentionedUsers	The usernames of the users mentioned in the tweet
mentionedHashtags	The hashtags mentioned in the tweet
inReplyToUserId	The user id of the user that the tweet is sent to reply
inReplyToTweetId	The tweet id of the tweet is sent to reply
rtTweetId	The tweet id of the original tweet that is retweeted
isRt	The flag that represents if the tweet is a retweet
publishDate	The date the tweet is posted

[1] http://twitter4j.org/en/index.html

Table 2. The features extracted from the owners of the collected tweets through the Twitter API *(Twitter API ile toplanan tweetlerin sahiplerin özellikleri)*

Feature	Description
userId	The id of the Twitter user
username	The username of the Twitter user
name	The name of the Twitter user
tweetCount	The number of tweets that the Twitter user has posted
followersCount	The number of follower users of the Twitter user
followingCount	The number of following users of the Twitter user
likeCount	The number of likes of the Twitter user
location	The location information of the Twitter user provided by him/her
description	The description of the user provided by him/her
website	The website of the user provided by him/her
isVerified	The verification status of the user's profile
listedCount	The number of lists the user has added
isDefaultProfileImage	The information about the user if he/she uses the default profile image
hasCustomBackground	The information about the user if the user's profile has a custom background
timeZone	The timezone information of the user
lang	The language of the user
creationDate	The date the user has created his/her profile

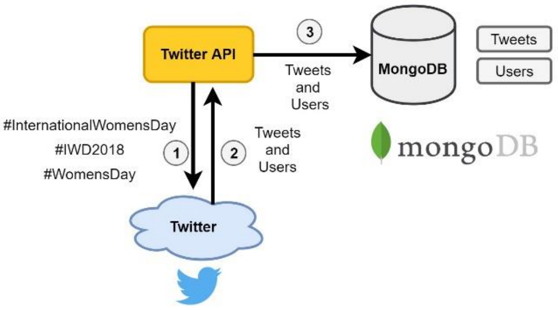

Figure 1. The process of data collection *(Veri toplama işlemi)*

3.2. Data Update Process *(Veri Güncelleme İşlemi)*

After the real-time tweets related to "International Women's Day" were fetched, a time interval was necessary in order to let tweets fulfill their impact on Twitter. This time interval was determined as one week which was thought to be long enough for a tweet to fulfill its impact on the Twitter network especially when it is considered that the trends (often called as Trending Topics - TT) of Twitter change daily if not hourly. After one week, a developed service was executed to update the tweets which were stored in a NoSQL database namely MongoDB[2]. Since the tweet ids were also stored in the database (see Table 1), the one-week later impact of each tweet has gained was retrieved through the Twitter API. The updated fields of each tweet were the number of likes each tweet has gained (favoriteCount) and the number of times each tweet was retweeted (rtCount). Since Twitter lets developers query 900 tweets by their ids per 15 minutes [33], the developed service waits 15 minutes between each execution. The whole data update process is presented in Fig. 2 as the (1) presents the process of Java service that firstly loads whole the stored tweets from MongoDB, (2) presents the process of Java service that extracts the tweetIds of the stored tweets, (3) presents the process of Java service that calls the related method of Twitter API with each loaded tweetId, (4) presents querying Twitter through the Twitter API, (5) presents the process of getting the tweet through the tweetId using the Twitter API, (6) presents the process of returning the tweet data from Twitter API, and (7) presents the process of Java service of updating the tweet data in MongoDB through the tweetId.

[2] https://www.mongodb.com

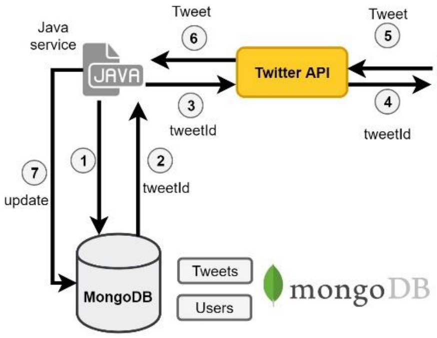

Figure 2. The process of data updates *(Veri güncelleme işlemi)*

3.3. Features Used for Classification *(Sınıflandırmada Kullanılan Özellikler)*

The features used with machine learning algorithms contain both the features extracted from Twitter API and the features constructed through the collected tweets as they are listed in Table 3. The Spread Score calculation is based on the gained retweet number of a user's tweets. Therefore, we did not use the gained retweet number of a user's tweets as a feature in order to not obtain biased results.

Table 3. The features used with machine learning algorithms (Makine öğrenmesi algoritmaları ile kullanılan özellikler)

Feature	Description
tweetCount	The number of tweets that the Twitter user has posted
followersCount	The number of follower users of the Twitter user
followingCount	The number of following users of the Twitter user
likeCount	The number of likes of the Twitter user
isVerified	The verification status of the Twitter user's profile
listedCount	The number of lists the Twitter user has added
numOfTweetsInDataset	The number of tweets of the Twitter user in the dataset
numOfRTTweetsInDataset	The number of retweeted tweets of the Twitter user in the dataset

3.4. Labeling Users as Influencer or Non-Influencer *(Kullanıcıların Etkin ve Etkin Olmayan Şeklinde Etiketlenmeleri)*

We need to measure the influence level of each user first so that we will be able to label which of the users in the data we obtain are influential. For this purpose, we have calculated the Spread Score [21] for all users. The second thing to do was to label users who have a higher score than a certain Spread Score as an influencer; labeling others as non-influencers. Empirically, we labeled the users who have Spread Score greater than $\text{mean} + 2 \times \text{standard\_deviation}$ as influencers (class 1), and we labeled the others as non-influencers (class 0). As a result, we have achieved 253 influencers and 167.915 non-influencers in our data set. At this point, as we can understand from the cardinality of the classes in the dataset, the dataset we obtained is the imbalanced dataset. Most classification methods work well with the balanced dataset (if the numbers of elements in the classes are close to each other) but they face challenges when the dataset classes are imbalanced. In this case, classification methods tend to be biased towards the majority class [34]. There are many resampling methods in the literature to overcome this problem [34]–[36]. One of the baseline methods is Random subsampling. Random subsampling removes randomly selected members from the majority class. We adopted Random subsampling to reduce the majority class (in this case non-influencers class) and we removed 90% of its members. We eventually obtained a reduced new training dataset, which consists of 253 influencer users and 16.792 non-influencer users. Non-influencer to influencer ratio in the new dataset may seem high. However, we did not make the dataset even smaller as we had good results in our experiments. In our experiments, we used the reduced dataset for training and the original dataset for testing.

3.5. Compared Classification Methods *(Karşılaştırılan Sınıflandırma Yöntemleri)*

Sure, it is impractical to make experiments with all classification methods and its parameters in an exhaustive manner. We compared many classification methods, which are widely used in the literature. These methods are Bayes Network Classifier, Hoeffding Tree [37], K-nearest neighbours classifier [38], J48 [39], KStar [40], Locally Weighted Learning [41], Multilayer Perceptron, Naive Bayes Classifier [42], NBTree

[43], Random Forest [44], Random Tree, and REPTree. We have used Weka as the machine learning tool [22].

The whole process of the proposed approach is presented in Fig. 3 as (1) presents the process of the implemented Java service that calculates the features from the updated data stored in the database, labels the users as either influencer or non-influencer, and exports the *.arff* files which are the files necessary to use the set within *Weka*, (2) presents the process of obtaining the training and test set, (3) presents the process of classification provided by *Weka*, and (4) presents the process of the classification of the test set as influencer candidates or non-influencer. The machine learning algorithms utilized in this study are based on the default configurations provided by *Weka*.

4. EXPERIMENTAL RESULTS (DENEYSEL SONUÇLAR)

In this section, the experimental result is described in each following subsection.

4.1. Classifying Users *(Kullanıcıların Sınıflandırılması)*

The results obtained for all classification methods are shown in Table 4. Here, NI stands for Non-Influencers class, and I stands for Influencer candidates class. For the evaluation, we compared the frequently used statistical measures. However, from the measures in Table 4, the measure that allows us to make the most healthful assessment for our problem is the MCC (Matthews Correlation Coefficient). For evaluating the performance of classification methods on an imbalanced dataset, inadequate performance metrics, such as accuracy, True Positive (TP), False Positive (FP) gives poor generalization results. MCC is widely used in such as situations [34], [45]. Since we are looking for a set of influential individuals in the IM problem, the key results that we should consider are (1) the multiplicity of the number of influencers that are classified as influencer candidates, and (2) the rarity of the number of non-influencers that are classified as influencer candidates. We cannot deduce that from the parameters like TP, FP, precision etc. Hence, the most successful method is Random Forest according to MCC measure. Also, we give the confusion matrices for all classification methods in Table 5. Random Forest correctly classified 145 out of 253 influencers, and only misclassified 75 non-influencers as influencer candidates.

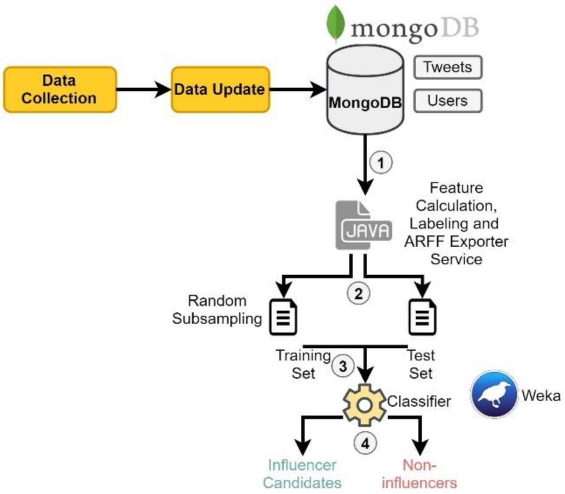

Figure 3. The whole process of the proposed approach *(Önerilen yaklaşımın bütün süreci)*

Table 4. Results of the classification methods *(Sınıflandırma yöntemlerinin sonuçları)*

Classifier	Class	TP Rate	FP Rate	Precision	MCC
BayesNet	NI	0,990	0,383	0,999	0,225
	I	0,617	0,010	0,084	0,225
HoeffdingTree	NI	**1,000**	1,000	0,998	-
	I	0,000	0,000	-	-
K-nearest neighbours KNN=1	NI	0,998	0,387	0,999	0,441
	I	0,613	0,002	0,319	0,441
K-nearest neighbours KNN=2	NI	**1,000**	0,763	0,999	0,339
	I	0,237	0,000	0,488	0,339
J48	NI	0,999	0,668	0,999	0,374
	I	0,332	0,001	0,422	0,374
KStar	NI	0,998	0,470	0,999	0,377
	I	0,530	0,002	0,271	0,377
LWL	NI	**1,000**	1,000	0,998	-
	I	0,000	0,000	-	-
Multilayer Perceptron	NI	**1,000**	0,881	0,999	0,219
	I	0,119	0,000	0,405	0,219
Naive Bayes	NI	0,995	0,684	0,999	0,159
	I	0,316	0,005	0,082	0,159
NBTree	NI	**1,000**	0,704	0,999	0,386
	I	0,296	0,000	0,503	0,386
RandomForest	NI	**1,000**	0,427	0,999	**0,614**
	I	0,573	0,000	0,659	0,614
RandomTree Seed=1	NI	0,998	0,387	0,999	0,422
	I	0,613	0,002	0,292	0,422
RandomTree Seed=2	NI	0,998	0,415	0,999	0,435
	I	0,585	0,002	0,325	0,435
REPTree	NI	0,999	0,636	0,999	0,326
	I	0,364	0,001	0,295	0,326

Here, the question of what is the quality of the influencers correctly classified can come to mind. We checked the classified users. We have seen that Random Forest method has correctly classified top 100 users, which have the highest influence.

Table 5. Confusion matrices for the utilized classification methods *(Kullanılan sınıflandırma yöntemlerinin hata matrisleri)*

			Predicated Class	
Classifier			**NI**	**I**
BayesNet	**Actual Class**	**NI**	166215	1700
		I	97	156
HoeffdingTree		**NI**	167915	0
		I	253	0
K-nearest neighbours KNN=1		**NI**	167584	331
		I	98	155
K-nearest neighbours KNN=2		**NI**	167852	63
		I	193	60
J48		**NI**	167800	115
		I	169	84
KStar		**NI**	167554	361
		I	119	134
LWL		**NI**	167915	0
		I	253	0
MultilayerPerceptron		**NI**	167871	44
		I	223	30
NaiveBayes		**NI**	167025	890
		I	173	80
NBTree		**NI**	167841	74
		I	178	75
RandomForest		**NI**	167840	**75**
		I	108	145
RandomTree Seed=1		**NI**	167540	375
		I	98	155
RandomTree Seed=2		**NI**	167607	308
		I	105	148
REPTree		**NI**	167695	220
		I	161	92

4.2. Selecting Top-k Influencers from Predicated Influencers Class *(Belirlenen etkin birey adayları sınıfından Top-k adet etkin bireyin seçilmesi)*

Classification of users as influencer candidates and non-influencers is the first part of the solution. After that, we should select top-k influencers from the predicated influencer candidates class. As we mentioned before, Random Forest (the best method in this study) correctly classified 145 out of 253 influencers, and only misclassified 75 non-influencers as influencer candidates. Totally, we have 220 influencer candidates in the predicated class. In the literature, there many different approaches and metrics to select top-k influencers. Some of these approaches can be listed as (1) selecting most followed user (MF), (2) ranking users with PageRank and selecting top-k ranked user etc. [20], [21]. In this study, we have used MF. Also, we have proposed a new modified version of MF, named as MF Natural Person based on our observations on the influencers. MF Natural Person can be defined as choosing the ones belonging to natural persons from MF accounts. To evaluate the results of our classification approach, we tested the MF and MF Natural Person metrics on the entire data set and on the Influencer candidates class predicted by Random Forest. We named the experiments as MF on entire dataset, MF on predicated I-class, and MF Natural Person on predicated I-class. Predicated I-class means the Influencer candidates class predicted by Random Forest. Figure 4 and Figure 5 show the performances of the metrics for the different numbers of top-k users. As seen from the figures, the choice of top-k influencers among the predicated class gave much better results.

Here, we want to explain why we do not use the PageRank algorithm (or any other heuristic metric) for evaluation. First, our main purpose in this study is to consider the IM problem as a classification problem, rather than developing a new user ranking metric. Additionally, we aimed to show that different classification algorithms can be successfully applied to the IM problem, and we aimed to compare the performance of existed user ranking metrics among the entire dataset and predicated class of influencer candidates. We carried out this for MF metric. Another metric can be used easily. Second, in order to be able to calculate PageRank, we must demonstrate the following relationships of users as a graph. As we aforementioned, Twitter does not give information about who retweets whom. To overcome this problem, in some studies, topical networks have been constructed by using Latent Dirichlet Allocation (LDA). On these networks, nodes denote users who posted on the topic and edges denote following relationship [21]. However, in order to create such a network, it is necessary to collect long-term data. This is possible in topical influence analysis studies because users can tweet/retweet during the year. International Women's Day, which we have studied, is a day-to-day issue and it is not possible to collect long-term data. If the followers of someone who tweeted about International Women's Day did not tweet/retweet about this topic, they won't enter the data set. For these reasons, it was not possible to obtain a graph that which we could calculate the real (or at least the near) PageRank of a person.

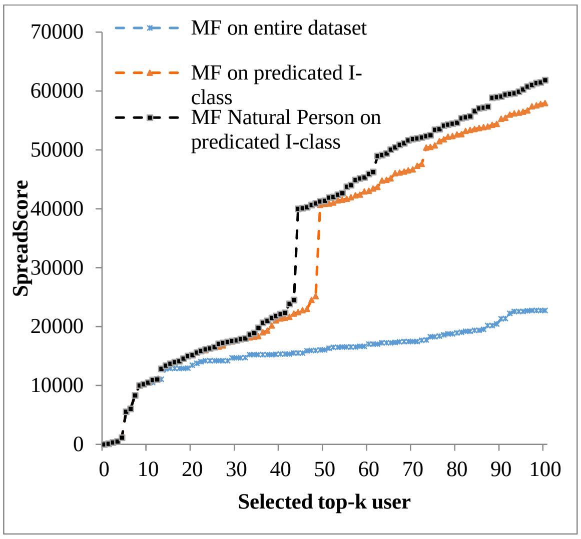

Figure 4. The calculated Spread Scores that shows the performances of the metrics for different number of top-k users *(Farklı sayıdaki top-k adet kullanıcı için farklı ölçütlerin performanlarını gösteren hesaplanmış SpreadScore değerleri)*

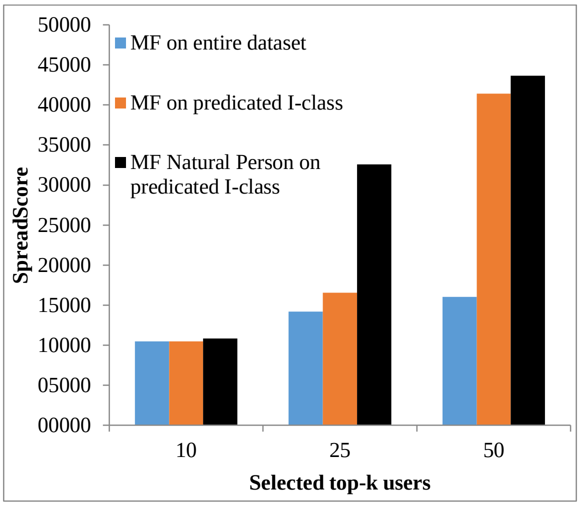

Figure 5. Performance of MF and MF Natural Person Heuristics for Different number of seeds *(Farklı çekirdek birey sayıları için MF ve MF Natural Person ölçütlerinin performansları)*

Lastly, we compared the performances of metrics with real (labeled) influencers. For this purpose, we selected real top-100 influencers and compared them with selected top-100 influencers by the metrics. As the result is presented in Fig. 6, the total *Spread Score* produced by all users is calculated as 171.603, the real top-100 influencers' *Spread Score* is calculated as 76.755, the *Spread Score* of *MF Natural Person on predicated I-class* is calculated as 61.814; the *Spread Score* of MF on predicated I-class is calculated as 57.919, and the *Spread Score* of *MF on the entire dataset* is calculated as 22.714.

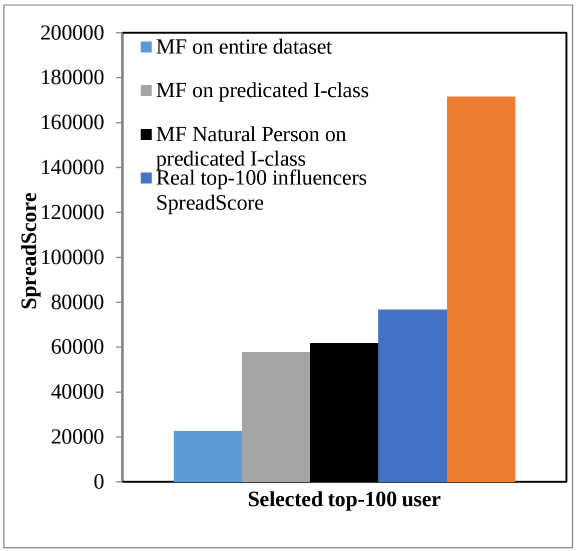

Figure 6. Comparison of the performance of the proposed method with real top influencers *(Önerilen yöntemin performansının gerçek etkin bireylerle kıyaslanması)*

The *Spread Score* of top-100 individuals by *MF Natural Person on predicated I-class* is calculated as 80,53% of the *Spread Score* of real top-100 most influential individuals.

5. RESULTS AND DISCUSSION *(SONUÇLAR VE TARTIŞMA)*

Experimental results show that the choices of top-k influencers made within the predicated influencer candidates class are better. We can think of the classification process as a reduction of the problem set. Thus, we have the chance to choose from a smaller number of individuals who are more likely to be influential. Also, since most of the classification methods work fast, this preprocessing step does not place a huge burden. For example, the Random Forest method, which gives the best result in this study, has spent 1-2 minutes to produce results for our data set with a standard laptop.

When the labeled data set is examined, we have seen that the influence levels of real accounts (owned by a natural person) are usually higher among MF accounts. Among the labeled (real) influencers, only 1 account in the first 10; 4 accounts in the first 25; 14 accounts in the first 50 are general accounts (not belonging to a real person). This is the reason why we proposed MF Natural Person heuristic. As it is stated by *Yang et al.* [46], the direct effect of individual-influence on MF upon the social media is stronger than that of the effect of an entity. Unsurprisingly, we have experienced that 46%

(23/50) of top 50 influencers are female as the topic we have studied on is directly related to female users. In order to process the vast amount of data fastly, the fetched tweets are stored in a NoSQL database management system namely *MongoDB* since it is reported that NoSQL databases provide better performance compared to relational databases especially when the size of data which is processed increases [47]–[50]. Similarly, the tweets fetched through the Twitter Streaming API need to be stored in the database with minimum delay in order to not miss any streaming tweets.

Known Limitations

This study's known limitations can be listed as follows:

- *Twitter Rate Limits.* Twitter defines rate limits for developers to query its data through the provided APIs. The rate limit (900 query per 15 minutes) decelerates the data update process enormously.
- *The necessity to update the stored tweets since the use of Twitter Streaming API.* The data update process is time-consuming especially when the number of tweets is huge.
- *Twitter API.* The Twitter API only gives the original tweet when the retweeted tweet is queried. Hence, the network of retweets could not be constructed. Another result of this limitation is that we could not compare the experimental result with the other related work based on a graph model such as *PageRank*.

6. CONCLUSION *(SONUÇ)*

In this paper, we propose the use of machine learning classification methods to detect influential users on Twitter. The proposed approach classifies users as influencer candidates and non-influencers with using user features and its tweets' attributes such as the number of tweets that the user has posted, number of follower users of the user etc. Thus, problem set is reduced. Selection of top-k influencers among influencer candidates (predicated class by the classifier) can be done according to a generic metric such as MF. Experimental results show that the selecting top-k influencers among predicated class outperforms the selecting among entire dataset. There are several users related and tweet related features on Twitter. We used a small subset of these features as mentioned in Section 3.3 to classify users. This ensures the classification of users in less time with fewer data.

We have labeled the users who have Spread Score greater than $\mathrm{mean} + 2 \times \mathrm{standart\_deviation}$ as influencer candidates (class 1), and we labeled the others as non-influencers (class 0). We empirically determined this threshold. A more detailed study can be done to determine the threshold value.

To obtain a sharp contrast between the classes, the number of target classes (which were defined as influencer candidates and non-influencers) may be increased, and the classified users as highest level influencer candidates may be selected as top-k influencers. Also, a hybrid metric may be developed by using the features that have a high positive impact on classification. As a future work, our approach can be applied on topical influencer analysis based on long-term data collection.

REFERENCES *(KAYNAKLAR)*

[1] M. Cha, H. Haddai, F. Benevenuto, and K. P. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy," in *International AAAI Conference on Weblogs and Social Media 2010 (ICWSM-10)*, 2010, pp. 10–17.

[2] L. Cui *et al.*, "DDSE: A novel evolutionary algorithm based on degree-descending search strategy for influence maximization in social networks," *J. Netw. Comput. Appl.*, vol. 103, no. September 2017, pp. 119–130, 2018.

[3] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '01*, 2001, pp. 57–66.

[4] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '03*, 2003, p. 137.

[5] D. Li, C. Wang, S. Zhang, G. Zhou, D. Chu, and C. Wu, "Positive influence maximization in signed social networks based on simulated annealing," *Neurocomputing*, vol. 260, pp. 69–78, 2017.

[6] L. Liu, J. Tang, J. Han, and S. Yang, "Learning influence from heterogeneous social networks," *Data Min. Knowl. Discov.*, vol. 25, no. 3, pp. 511–544, 2012.

[7] J. S. More and C. Lingam, “A SI model for social media influencer maximization,” *Appl. Comput. Informatics*, 2017.

[8] M. Richardson and P. Domingos, “Mining knowledge-sharing sites for viral marketing,” in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’02*, 2002, p. 61.

[9] Y. Zeng, X. Chen, G. Cong, S. Qin, J. Tang, and Y. Xiang, “Maximizing influence under influence loss constraint in social networks,” *Expert Syst. Appl.*, vol. 55, pp. 255–267, 2016.

[10] F. Li and T. C. Du, “Maximizing micro-blog influence in online promotion,” *Expert Syst. Appl.*, vol. 70, pp. 52–66, 2017.

[11] S. P. Borgatti, “Identifying sets of key players in a social network,” *Comput. Math. Organ. Theory*, vol. 12, no. 1, pp. 21–34, Apr. 2006.

[12] W. Chen, Y. Wang, and S. Yang, “Efficient influence maximization in social networks,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’09*, 2009, p. 199.

[13] T. Lappas, E. Terzi, D. Gunopulos, and H. Mannila, “Finding effectors in social networks,” in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’10*, 2010, p. 1059.

[14] J.-R. Lee and C.-W. Chung, “A Query Approach for Influence Maximization on Specific Users in Social Networks,” *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 2, pp. 340–353, Feb. 2015.

[15] D. Li, Z.-M. Xu, N. Chakraborty, A. Gupta, K. Sycara, and S. Li, “Polarity Related Influence Maximization in Signed Social Networks,” *PLoS One*, vol. 9, no. 7, p. e102199, Jul. 2014.

[16] S. Peng, A. Yang, L. Cao, S. Yu, and D. Xie, “Social influence modeling using information theory in mobile social networks,” *Inf. Sci. (Ny).*, vol. 379, pp. 146–159, Feb. 2017.

[17] K. Zhang, H. Du, and M. W. Feldman, “Maximizing influence in a social network: Improved results using a genetic algorithm,” *Phys. A Stat. Mech. its Appl.*, vol. 478, pp. 20–30, Jul. 2017.

[18] S. Peng, Y. Zhou, L. Cao, S. Yu, J. Niu, and W. Jia, “Influence analysis in social networks: A survey,” *J. Netw. Comput. Appl.*, vol. 106, no. November 2017, pp. 17–32, 2018.

[19] M. Samadi, R. Nagi, A. Semenov, and A. Nikolaev, “Seed activation scheduling for influence maximization in social netw
orks,” *Omega*, vol. 77, no. June 2018, pp. 96–114, 2018.

[20] J. V. Cossu, V. Labatut, and N. Dugué, “A review of features for the discrimination of twitter users: application to the prediction of offline influence,” *Soc. Netw. Anal. Min.*, vol. 6, no. 1, 2016.

[21] Z. Zengin Alp and S. Gunduz Oguducu, “Identifying topical influencers on twitter based on user behavior and network topology,” *Knowledge-Based Syst.*, vol. 141, pp. 211–221, 2018.

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software,” *ACM SIGKDD Explor. Newsl.*, vol. 11, no. 1, p. 10, Nov. 2009.

[23] M. Kitsak *et al.*, “Identification of influential spreaders in complex networks,” *Nat. Phys.*, vol. 6, no. 11, pp. 888–893, Nov. 2010.

[24] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, “Influence and passivity in social media,” in *Proceedings of the 20th international conference companion on World wide web - WWW ’11*, 2011, p. 113.

[25] M. Cataldi and M. A. Aufaure, “The 10 million follower fallacy: audience size does not prove domain-influence on Twitter,” *Knowl. Inf. Syst.*, vol. 44, no. 3, pp. 559–580, 2015.

[26] A. Pal and S. Counts, “Identifying topical authorities in microblogs,” *Proc. fourth ACM Int. Conf. Web search data Min. - WSDM ’11*, p. 45, 2011.

[27] H. Kwak, C. Lee, H. Park, and S. Moon, “What is Twitter, a social network or a news media?,” in *Proceedings of the 19th International Conference on World Wide Web (WWW’10)*, 2010, vol. 112, no. 2, pp. 591–600.

[28] J. Weng, E. P. Lim, J. Jiang, and Q. He, “Twitterrank: Finding topic-sensitive influential twitterers,” *Proc. 3rd ACM Int. Conf. Web Search Data Min. (WSDM 2010)*, pp. 261–270, 2010.

[29] A. Leavitt, E. Burchard, D. Fisher, and S. Gilbert, “The Influentials: New Approaches for Analyzing Influence on Twitter,” 2009. [Online]. Available: http://www.webecologyproject.org/wp-content/uploads/2009/09/influence-report-final.pdf. [Accessed: 05-Apr-2018].

[30] P. Ficamos and Y. Liu, “A Topic based Approach for Sentiment Analysis on Twitter Data.,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 12, pp. 201–205, 2016.

[31] P.-C. Lin and P.-M. Huang, “A Study of Effective Features for Detecting Long-surviving Twitter Spam Accounts,” in *2013 15th International Conference on Advanced Communications Technology (ICACT)*, 2013, pp. 841–846.

[32] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas, “Sentiment analysis leveraging emotions and word embeddings,” *Expert Syst. Appl.*, vol. 69, pp. 214–224, 2017.

[33] “Rate limits — Twitter Developers,” *Twitter*, 2018. [Online]. Available: https://developer.twitter.com/en/docs/basics/rate-limits.html. [Accessed: 05-Apr-2018].

[34] S. Boughorbel, F. Jarray, and M. El-Anbari, “Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric,” *PLoS One*, vol. 12, no. 6, p. e0177678, Jun. 2017.

[35] A. More, “Survey of resampling techniques for improving classification performance in unbalanced datasets,” vol. 10000, pp. 1–7, 2016.

[36] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, and J. M. Pérez, “Applying Resampling Methods for Imbalanced Datasets to Not So Imbalanced Datasets,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8109 LNAI, 2013, pp. 111–120.

[37] G. Hulten, L. Spencer, and P. Domingos, “Mining time-changing data streams,” in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’01*, 2001, pp. 97–106.

[38] D. W. Aha, D. Kibler, and M. K. Albert, “Instance-based learning algorithms,” *Mach. Learn.*, vol. 6, no. 1, pp. 37–66, Jan. 1991.

[39] S. L. Salzberg, “C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993,” *Mach. Learn.*, vol. 16, no. 3, pp. 235–240, Sep. 1994.

[40] J. G. Cleary and L. E. Trigg, “K*: An Instance-based Learner Using and Entropic Distance Measure,” in *Proceedings of the Twelfth International Conference on International Conference on Machine Learning*, 1995, pp. 108–114.

[41] C. G. Atkeson, A. W. Moore, and S. Schaal, “Locally Weighted Learning,” *Artif. Intell. Rev.*, vol. 11, no. 1–5, pp. 11–73, 1997.

[42] G. H. John and P. Langley, “Estimating Continuous Distributions in Bayesian Classifiers,” in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, 1995, pp. 338–345.

[43] R. Kohavi, “Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-tree Hybrid,” in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 202–207.

[44] L. Breiman, “Random Forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[45] B. W. Matthews, “Comparison of the predicted and observed secondary structure of T4 phage lysozyme,” *Biochim. Biophys. Acta - Protein Struct.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.

[46] Y. Yang, S. Hung, Y. Zhang, and Y. Shen, “INDIVIDUAL-INFLUENCE & INTERACTIVE-RELATIONSHIP : THE APPLICATION UPON SOCIAL MEDIA FOR PUBLISHING-RELATED,” in *Forty-Sixth Annual Meeting of the Western Decision Sciences Institute (WDSI 2017)*, 2017, pp. 1–6.

[47] D. Bartholomew, “SQL vs. NoSQL,” *Linux J.*, vol. 2010, no. 195, pp. 54–59, 2010.

[48] Y. Li and S. Manoharan, “A performance comparison of SQL and NoSQL databases,” in *IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing - Proceedings*, 2013, pp. 15–19.

[49] Jing Han, Haihong E, Guan Le, and Jian Du, “Survey on NoSQL database,” in *2011 6th International Conference on Pervasive Computing and Applications*, 2011, pp. 363–366.

[50] A. Boicea, F. Radulescu, and L. I. Agapin, “MongoDB vs Oracle - Database comparison,” in *Proceedings of 3rd International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2012*, 2012, pp. 330–335.

Mehmet ŞİMŞEK

Mehmet Şimşek received the bachelor’s degree in computer engineering from Selçuk University in 2004, He received master’s degree in computer engineering from Gazi University in 2006, and philosophy of doctorate degree in electronics and computer education from Gazi University in 2012. Currently, He is working as assistant professor at Düzce University, Faculty of Engineering, Department of Computer Engineering. His current research interests include graph theory, large and complex networks, social networks, machine learning techniques for big data analytics and Internet of things security.

Abdullah Talha KABAKUŞ

A. Talha Kabakuş received the bachelor’s degree in computer engineering from Çankaya University in 2010, the master’s degree in computer engineering from Gazi University in 2014, and the philosophy of doctorate degree in Electrical-Electronics & Computer Engineering from Düzce University in 2017, respectively. He is currently working as assistant professor at Faculty of Engineering, Department of Computer Engineering, Düzce University. His research areas include mobile security, sentiment analysis, natural language processing, big data analysis, and learning systems.