Social Networking, 2013, 2, 165-173
http://dx.doi.org/10.4236/sn.2013.24016 Published Online October 2013 (http://www.scirp.org/journal/sn)

Recommending Who to Follow on Twitter Based on Tweet Contents and Social Connections

Evgenia Tsourougianni, Nicholas Ampazis
Department of Financial and Management Engineering, University of the Aegean, Chios, Greece
Email: fmem09029@fme.aegean.gr, n.ampazis@fme.aegean.gr

Received July 10, 2013; revised August 14, 2013; accepted September 30, 2013

Copyright © 2013 Evgenia Tsourougianni, Nicholas Ampazis. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT

In this paper, we examine methods that can provide accurate results in a form of a recommender system within a social networking framework. The social networking site of choice is Twitter, due to its interesting social graph connections and content characteristics. We built a recommender system which recommends potential users to follow by analyzing their tweets using the CRM114 regex engine as a basis for content classification. The evaluation of the recommender system was based on a dataset generated from real Twitter users created in late 2009.

Keywords: Recommender Systems; Social Networks; Personalization

1. Introduction

Twitter and other social networks (Facebook, Digg, Flickr, MySpace, etc.) enable users to realize the value of interacting with other people, communicate with no geographic constraints, and exchange opinions, ideas and sentiment, having the great advantage of low barriers to participation. They actually represent social interactions and can be used to study information and ideas diffusion and social bonds. Twitter, in particular, combines elements of social networking sites and microblogging services (microblogs broadcast information about one's activities, opinion and status). Twitter is a microblogging service which was founded in July 2006 to enable people to share short textual messages with each other (called tweets) [1]. It is a social network as well, used by friends, family and co-workers to communicate and stay connected through the exchange of quick, frequent answers to one simple question: "What are you doing?" [2]. Twitter is one of the most popular web services of the Web 2.0 era. It maintains an exponential growth, since by September 2008 it reached 3 million users, in July 2009 it reached 41 million users [2], eventually reaching the current number of 75 million users, of which, 10 - 15 million are active users [3].

Today, Twitter is used by many people as a form of a news reader, as users follow their favorite news sources. It has also proven to be a very popular way for sharing interesting content discovered on the Web among the circle comprising the social connections of users (otherwise known as "social graph"). However, the transformation of users to information producers and the continuous update of a user's tweet stream (*i.e.* the tweets of the other users that he follows), rapidly increases information overload and renders more difficult the discovery of interesting content. In addition, interesting content might reside outside of a given user's tweet stream. Discovering potential users to follow which may provide such interesting content (among the millions of Twitter users) is indeed a very challenging task.

In this paper, we consider the problem of recommending Twitter users to follow based on a user modeling perspective. We adopt a content-based strategy, which relies on the content of the users' tweets as well as the content of the tweets of the users that they follow, in order to create profiles that can be utilized for expanding their social graph. Since Twitter limits the length of a tweet to only 140 characters, and users may tweet on a variety of different topics, traditional Information Retrieval (IR) and text classification strategies are likely to produce dubious results. To this end, we utilize the CRM114 Controllable Regex Mutilator [4], which in applications involving noisy datasets (such as spam filtering), has been shown to be as highly as 99.9% accurate [5]. CRM114 is used in order to train on the user tweet streams, and to classify tweets outside of a given user's tweet stream (by expanding the user's social graph). The sources of interesting tweets are consequently recommended as potential new users to follow. In order to evaluate the efficiency of this approach we perform an off-line recommendation evaluation using real-world data collected from approximately 60,000 Twitter users in late 2009.

The rest of the paper is organized as follows: First, we discuss related research to our work. We then provide an overview of social interactions on Twitter and our user modeling approach. Then, we describe our recommender strategy, and consequently detail our case study and experimental results. The paper ends with conclusions drawn and possible generalizations of our approach to related recommendation problems.

2. Related Work

There is a great range of research upon Twitter, conclusions of which have been used in order to analyze social interactions at network and social levels [6], to explore the social and behavioral consequences of its usage [7], to answer questions concerning social behavior [8] or even to analyze the influence of Twitter on politics [9]. In addition, they've been used to analyze and discover latent characteristics of users [10] or even analyze the syntax of retweets, how and why people retweet and what they retweet.

Daniel Tunkelang proposed the TunkRank algorithm in order to measure the influence of Twitter users [11]. Cha *et al*. proposed other metrics to count influence and supported the idea that followers don't equal influence [12]. Also, Gayo-Avello and Brenes evaluated the performance of 5 different ranking algorithms: PageRank, HITS, NodeRanking, TunkRank and TwitterRank for separating most relevant users from spammers [13].

Recommender systems that utilize content are more frequently used in domains where extensive textual content is available. Examples include recommendations for items such as websites [14] and books [15]. Other studies have investigated the structure of social networks for providing recommendations for friends-of-friends [16, 17], while there are also hybrid approaches which combine collaborative filtering and content-based methods, e.g. [18].

The study which is more closely related to our work is by Hannon *et al*. [19] on the recommendation of users to follow using content and collaborative filtering approaches. In that study the authors describe a number of different user profiling strategies in order to learn about the interests of individual users. The authors also provide details on the implementation of a system called "Twittomender"[1], which provides recommendations for potentially interesting users to follow. An also, very closely, related study to our work is by Chen *et al*. [20] on the development and evaluation of URL recommendation strategies using various combinations of tweet content and social graph information. They also describe the design and empirical studies of a recommender system built on top of Twitter, called "zerozero88"[2], which recommends URLs that a particular Twitter user might find interesting.

The main contribution of this paper is that our work differs from the approaches describe above in two important ways:

- The first is that, we do not represent content based on the frequently used Vector Space Model (VSM), which treats queries and documents as vectors of individual words, and computes their similarity as the deviation of angles between them. Instead, we consider representing content by utilizing spam filtering features, namely Orthogonal Sparse Bigrams (OSB), in order to construct rich feature sets which can effectively model the noisy and unstructured user tweets.
- Secondly, we view the recommendation process as a classification rather than a retrieval problem, in order to classify tweets of potentially interesting users to follow. To the best of our knowledge this approach has never been applied before in the framework of social recommendations.

In addition, similarly to the work of [19], we demonstrate that, even though Twitter data are noisy, it is still possible to extract useful signals for providing recommendations.

3. Modeling Users on Twitter

On Twitter a user A has the ability to follow another user B (without his approval). This means that A is able to read B's tweets. In this relationship user B is defined as user's A *followee (or follow)* and user A is defined as user's B *follower*. When there is reciprocity in a relationship, (*i.e.* user A follows user B and user B follows user A), these two users are usually denoted as *friends*.

Users on Twitter have different strategies for deciding who to follow. They follow other users, not only because they may be personal friends with them in real life, but also because they may be interested in what they say (even if they are complete strangers or perhaps their followee is a celebrity). Sometimes they even follow others in hope of reciprocity [21]. Some users may also be abusing Twitter in order to get "graph prestige" [13]. However, in contrast with other social networks, on Twitter, relationships are not always mutual. Studies have shown that only 22.1% of users follow each other [2]. As a matter of fact, following a user is usually not reciprocated. Surprisingly, 67.6% of users are not fol-

[1] http://twittomender.ucd.ie

[2] http://zerozero88.com

lowed by any of the users they follow. In addition, it has been shown that users with less than 10 followers almost never tweet, but there are users that tweet more often than expected considering their number of followers [2].

In this section, we will look at some different sources of content information available for profiling users. Our aim is to provide recommendations for Twitter users in order to help them create interesting Twitter streams. Specifically, we seek a way to utilize their existing set of connections, and to recommend interesting *followees-of-followees* (*fof*) for them to follow. This means that we wish to provide recommendation for users to follow from the pool of the people that are already followed by a user's own followees. This would result in expanding their social graph to one additional level. In social networking theory, these are known as *friends of friends*, and it has been shown that the degree of influence drops off significantly after this tier of connections [22].

Obviously, the simplest source of profiling information is the set of users' *own* tweets. This set can provide us with a basis for a content-based approach to user profiling, assuming, of course, that users usually tweet about things that interest them. Another potential source of profiling information is the tweets of a user's *followees*. Since the decision to follow a certain user is a deliberate action, we may safely assume that users expect their folowees' tweets to be of interest, and therefore interpret this as a preference signal. On the contrary, since, on Twitter, users do not have any control over who follows them, we expect that considering *follower* accounts would probably not yield relevant information about a target user's preferences. This assumption is further supported by the recent explosive growth in the number of "bot" accounts that seek to follow as many users as possible in the hope that unaware users will follow them back, so as to increase their ranking [12].

In summary, the above discussion suggests two basic profiling strategies for modeling users on Twitter based on content:

1) Users are represented by their *own* tweets;

2) Users are represented by a mixture of their *own* tweets and of the *union* of the tweets of their *followees.*

Using these two approaches as a basis for profiling Twitter users, we can then process these profiles and develop the recommendation framework to deliver results based on target *fof* profiles as described in the following two sections.

4. Feature Extraction

There are many design choices when deciding upon feature extraction for text classification given a corpus to work with. To break an incoming text up into tokens, one can use any regular expression (regex) and each successive non-overlapping match of the regex extracts a token. The usual choice for the regular expression is the (POSIX-format) regex [[:graph:]]+ which produces a stream of variable-length real language words. This regex creates the traditional bag-of-words text representation and is a baseline modeling where texts are represented by counts of the words that they contain [23].

As already mentioned, Twitter limits the length of a tweet to only 140 characters. On one hand, this limit is beneficial for users since it makes easier the scanning of their Twitter stream, helping them skip over tweets that are not very interesting (short tweets are important to reading as well). However, this limit may also be an inherent source of noise in the tweets, since users are inclined to use various word abbreviations and emoticons, commonly encountered in SMS messages. In addition, tweets may contain links to URLs which are usually represented by short sequences of random characters generated by URL shortener services, such as tinyurl.com, bit.ly, goo.gl and others.

In order to mimimize the noise in our real word tokens, in our implementation, we cut off the URLs existing in the tweets, and also cut off all the tweets that contained replies to certain users (identified by the presence of the "@" character in the beginning of a tweet). The reason for the latter decision is to further reduce noise since these tweets are directed to a certain user and are equivalent to "chats" between two users on Twitter. However, we kept all retweets, (*i.e.* echoes of a followee's tweet to the user's own tweet stream) and tweets including "hashtags". Replies are identified by the presence of the "RT" characters in the beginning of a tweet, and hashtags are marked with the "#" symbol to add additional context and metadata to tweets in order to categorize them. A retweet can be considered as a positive signal of interest in a folowee's tweet, and sharing a hashtag implies that two tweets are related to the same topic.

Real word tokens from the filtered tweets, as described above, were combined as Orthogonal Sparse Bigrams (OSB) in order to construct feature sets for each tweet. The OSB feature set is described in [24], and has been experimentally verified to produce higher quality representations when compared to other feature sets [25]. OSB uses a word pairing of two words at a time in any given N-word window, and thus only $N - 1$ combinations with exactly two words are produced. For example, with a sequence of five words, w1, …, w5 OSB produces four combined features (tokens):

w1 w2
w1 <skip> w3
w1 <skip> <skip> w4
w1 <skip> <skip> <skip> w5

The OSB features form an almost complete basis set. All of the OSB features are unique and not redundant because it is not possible to obtain any OSB feature by

Copyright © 2013 SciRes.

adding, ORing, or subtracting any other pairs of other OSB features. Experiments have consistently shown that the use of OSBs increases classification accuracy compared to using *unigram* (*i.e.* single word) representations [24].

5. Classification Based Recommendations

Our approach to classification based recommendations was implemented using the CRM114 Controllable Regex Mutilator. CRM114 is a language suited to examine incoming data streams, and to sort, filter, or alter these streams according to criteria set by the programmer. CRM114 can be utilized to alert to important events but simultaneously ignore ones that aren't meaningful. CRM114 is described in depth in the manual CRM114 Revealed [4] and is available for free download from the CRM114 web page[3].

The most important statements implemented in CRM114 are the LEARN and CLASSIFY statements. LEARN and CLASSIFY can be utilized in order to train and test respectively, classifiers specified within the framework of the language. CRM114 comes with a variety of readily available classifiers embedded into the language such as Bayesian, Markovian, Winnow, and others. A detailed description of the preinstalled CRM114 classifiers may be found in [24,26].

In our experiments, we utilized the Markovian classifier using the OSB feature set described in the previous section. The Markovian classifier is an extension of Bayesian classification, which maps features in the input text into a Markov Random Field. This turns each token in the input into 2(N − 1) features where N is the total number of tokens. In case where tokens are simple words (*unigram*) the Markovian classifier reduces to the simple Naive Bayes classifier where a document d_i is assigned (among m categories) to category c_l if where a document di is assigned (among m categories) to category c_l if

$$\begin{aligned} l &= \arg\max_{j=1,\cdots m} P\left(c_j \middle| d_i, \theta\right) \\ &= \arg\max_{j=1,\cdots m} P\left(c_j \middle| \theta\right) P\left(d_i \middle| c_j, \theta\right) \end{aligned} \tag{1}$$

The likelihoods $P\left(d_i \mid c_j, \theta\right)$ are computed using the (naive) independence assumption that

$$P\left(d_i \middle| c_j, \theta\right) = \prod_{k=1}^{|d_i|} P\left(t_k \middle| c_j, \theta\right) \tag{2}$$

where $t_1, \cdots, t_{|di|}$ are (in the Markovian case) combinations of the OSB tokens of the document d_i. The parameters θ are estimated from the training set, usually using a multinomial or a multiple Bernoulli model.

For each target user profile created as described in Section 3, we utilized the LEARN construct to create a model and utilized the CLASSIFY statement in order to count the percentage of the user's *known followees* tweets classified as positive; in other words, we look to see how often the recommender classifies as positives the tweets of the people that the target user is known to have already followed. Therefore, as in [19], our basic measure of recommendation performance is the average percentage overlap between a given recommendation list and the target users *actual* followees list.

6. The Classification Based Recommendations Architecture

The proposed recommendation system can be developed as a Web service, as shown in **Figure 1**. Through the Twitter API[4], it is possible to query Twitter for user IDs, their tweets, and their social graph (*i.e.* their followers and their followees).

In order to recommend users to follow for a particular user ID, the user's own Twitter profile acts as a form of query to the system. The Twitter profile of each user consists of the user's own Tweets, the user's social graph and the user's timeline (*i.e.* the tweets of his/her followees). For each user ID, an optimal classifier is trained (see Section 7) either on the user's own tweets or on the user's tweets and the union of the tweets of the user's followees. Then, in order to recommend new users to follow, from the expanded social graph of the user, we locate the followees for each one of the target user's followees (fofs), making sure that there is no fof that the target user already follows. We then select all of the fofs' tweets for classification, and we recommend those fofs that have the highest percentage of positively classified tweets (as shown schematically in **Figure 1**).

In the section that follows we will describe in detail the dataset used for our experiments and the results across the different user profiling strategies, focusing on the accuracy of the classifier model on the comparison between recommendation lists and actual followees lists.

7. Dataset and Experimetal Procedure

The dataset we chose to work with is a crawl of Twitter data performed in late 2009[5]. The crawl was seeded from a set of authoritative users, celebrity users (politicians, musicians, environmentalists, technology influencers, etc.) featured on the social media blog Mashable[6]. The dataset contained a file with tweets from 62,438 users, with a total number of tweets of about 2.3 million. The tweets spanned a period between April 2006 and December 2009 but they were not uniformly sampled, as most of the tweets were posted during 2009. In addition to the tweets data, a social

[3] http://crm114.sourceforge.net

[4] https://dev.twitter.com/

[5] http://www.public.asu.edu/~mdechoud/datasets.html

[6] http://mashable.com/2008/10/20/25-celebrity-twitter-users

Copyright © 2013 SciRes.

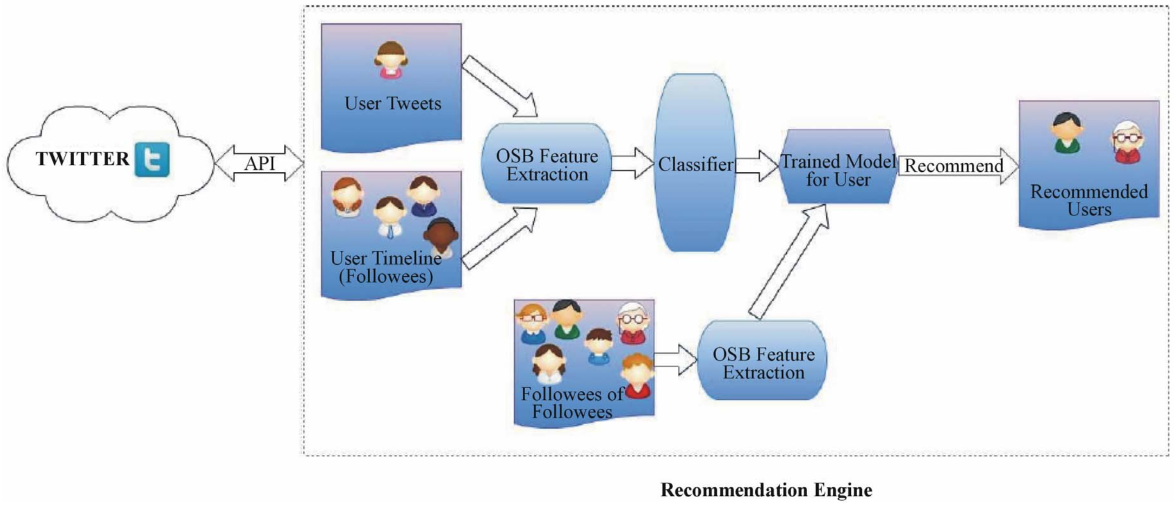


Figure 1. Schematic diagram of the recommendation architecture.

graph file was provided which depicted user/followees relations for 2503 out of the total 62,438 users. In order to contact our experiments, we worked on two subsets of the tweets of those 2503 users as described in the following two subsections. To protect user privacy, in our results we do not show real Twitter usernames. We also did not utilize any other information provided in the dataset apart from tweet contents and the user/followees relations. We must also mention that we imposed no limits on the time the tweets were posted (for example, tweets only posted during 2009, more recent tweets, etc.) on any of the tweets that we selected.

7.1. Experiments with the First Subset

We started by calculating the average number of followees among the 2503 users provided in the social graph file (*average.nu.followees* = 336). As a starting point, we selected target users that had a number of followees which was close to this average (*average.nu.followees* ± 10). We ordered them by the number of followees and we chose the top 10 users that had the largest number of tweets. For each one of those 10 users, we selected 10 followees—again those had the largest number of tweets, in order to construct a representative corpus to train and test our CRM114 classifier on.

Our first experiment aimed to verify the ability of our combination of methodologies for feature extraction and classification to produce meaningful results. To this end, we used 90% of each target user's own tweets for training and the remaining 10% for classification. The results of the classifications for all 10 target users are shown in **Figure 2**. From this figure, we can see that for all target users, at least 80% of their validation tweets were classified as positive, which means that the classifier was able to recognize the fact that a target user's own tweets were interesting to him/her.

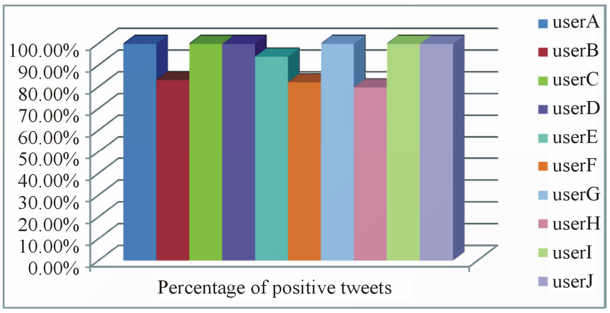


Figure 2. Classification result for target users' own tweets.

In our second experiment, for each target user, we used all of his/hers tweets to train the classifier and the tweets for each of his/her *known* followees for classification. For each target user's followee, we counted the percentage of the tweets that was classified as positive. High percentages would indicate that users find their followee's tweets interesting. The results of this trial are shown in **Figure 3**. For each subgraph in this figure, the horizontal axis is an index of the user's known followees, and the vertical axis is the percentage of tweets classified as positive for each followee. From this figure, we can see that, for most target users, more than 70% of their followees' tweets were classified as positive and hence they would worth been recommended to them as potential users to follow in a recommendation list. A high overlap between this list and a target user's actual followees-list would illustrate the efficiency of such an approach (see subsection 7.3). It is worth noting "userB" which is the only one that seems to exhibit relatively low interest on his/her followees' tweets. A possible explanation to this might be that either userB randomly chooses followees (e.g. bot) or that he/she does not share much the same interests with his/her followees.

We then tried to locate 10 followees for each one of our 10 target user's followees (fofs), in order to expand their

Copyright © 2013 SciRes.

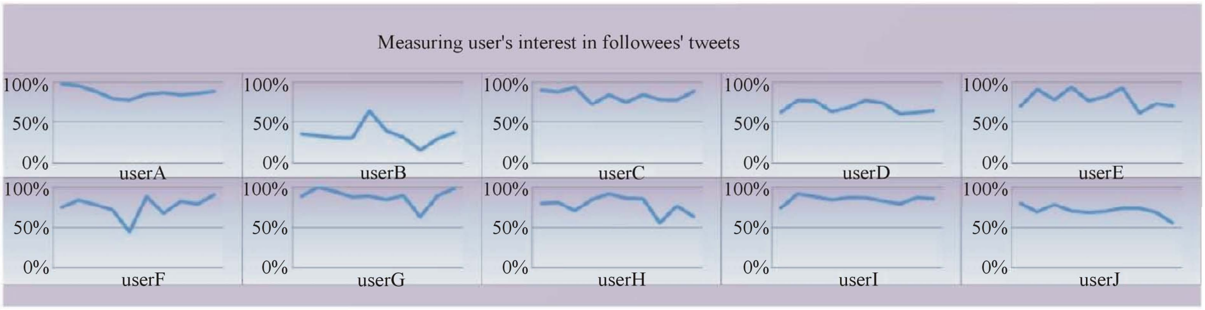


Figure 3. Percentage of positively classified tweets for all target users for all of their followees.

social graph and test our approach in recommending fof users to follow. Unfortunately, this proved infeasible since we could not locate any fofs in the downloaded dataset, that is, none of the 100 folowees of our initial 10 target users had any further connections in the social graph file. To this end, we reproduced our experiments using a different seed of initial target users as described in the following subsection.

7.2. Experiments with the Second Subset

For the construction of this candidate set, as target users we selected 10 users which met three conditions: 1) they were following "authority" users, *i.e.* users with the largest number of followers; 2) their followees had more than 100 tweets; and 3) at least one of their "authority" followees was following a celebrity user from which the dataset was originally seeded. We were not so firm in condition 2), however, due to the limited number of available tweets in the dataset for some of them. Finally, for each one of the followees, we selected at most 10 of their followees (fofs), provided that they had enough tweets.

As in our first experiment with the first candidate set, we tried to verify the ability of the classifier to classify correctly those new target users' own tweets. Again we used 90% of each target user's own tweets for training and the remaining 10% for classification. The results of these classifications for all 10 target users are shown in **Figure 4**. From this figure we can see that for almost all target users, the majority of their validation tweets were classified as positive, which implies that a user is interested in the content of his own tweets, a fact which is captured by the CRM114 classifier.

In our second experiment with this set, for a target user, we used all of his/her tweets for training the classifier and evaluated the classification on all the tweets posted by each of his/her *known* followees. The results of this experiment are shown in **Figure 5**. The notation $follow_iX$ implies that user X is a followee of user i. The high percentages illustrated in this figure imply that users are indeed interested in the contents of their followees' tweets. For most target user's followees, more than 80% of their tweets were classified as positive which means that they would rank highly in a top-N recommendation list. Since these followees are known to be the target user's actual followees, again this illustrates the efficiency of this approach in recommending users to follow (see also subsection 7.3). We should also make a comment on the fact that user8 exhibited 0% in the classification of his followees tweets. Intrigued by this fact we closely inspected his/her tweets used to train the CRM114 classifier, and found that he/she always uses the exact same pattern in the posted tweets, which is completely unrelated to the tweets of his/her followees (probably a bot).

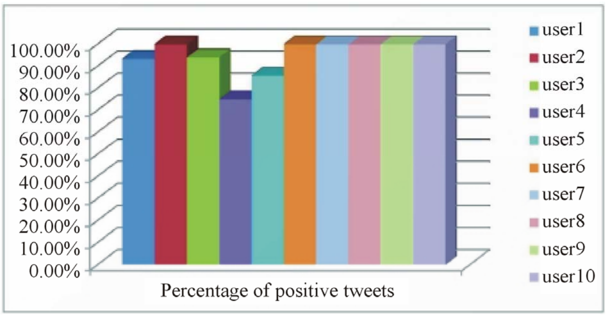


Figure 4. Classification result for target users' own tweets.

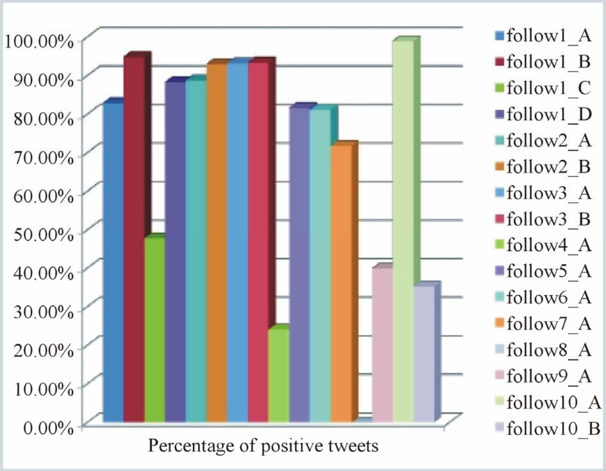


Figure 5. Percentage of positively classified tweets for all target users for all of their followees.

In order to recommend new users to follow from the expanded social graph of the user, we located 10 followees for each one of the target user's followees (fofs), having made sure that there was no fof that the target user already followed. We selected all of the user's tweets for training and we classified the tweets for each one of the fofs. As a threshold, we set 75% of positive tweets, *i.e.* if the fof had more than 75% of his/her tweets classified as positive, then he/she should be recommended to the user as an interesting user to follow. The results of this trial are shown in **Figure 6**. Again, for each subgraph in this figure, the horizontal axis is an index of the user's fofs, and the vertical axis is the percentage of tweets classified as positive for each fof. The overall results show that almost 62% of the fofs could be recommended to the initial users as potentially interesting users to follow (only user8's fofs had 0% of positive tweets for the reason mentioned previously). This should be viewed as a very positive result since the limited number of tweets available in the dataset could've had potentially degraded the performance of the CRM114 classifier, which was nevertheless able to capture the existence of similar interests in content among target users and fofs.

In our final experiment with this subset, we repeated the previous setup, but this time we trained the classifier with all of the user's tweets and the *union* of the tweets of the user's followees, and we then classified each of the fofs' tweets. As a threshold, we again set 75%, *i.e.* if the fof had more than 75% of his/her tweets classified as positive, then he/she should be recommended to the user as a potentialy interesting user to follow. The results of this trial are shown in **Figure 7**. As usual, the horizontal axis is an index of the user's fofs, and the vertical axis is the percentage of tweets classified as positive for each fof. The results reveal an increase in the percentage of tweets that in this case are classified as positive, since the enhanced representation of users as a mixture of their own tweets and the tweets of their followees increased the average percentage from 62% (previous experiment) to almost 80%. In this case, even user8's fof had an average of 50% of their tweets classified as positive due to the beneficial inclusion of his/her's own followees in the user modeling. Presumably, this is a further indicator of a consistent positive signal among the interests within the 3-tier user/followee/fof chain.

7.3. Recommendation Precision and Ranking

In order to further quantify our results, as a recommendation performance measure we considered the average percentage overlap between a given recommendation list and the target users *actual* followees-list. For each trained classifier model corresponding to our target users (considered in both subsets), we classified the tweets of every possible followee, regardless of the fact that the followee was indeed an actual followee of the target user or a followee of another user. We then ordered the results among all the followees according to the percentage of the tweets classified as positive and, for different recommendation list sizes (k), we counted how many of the possible recommendations are in the users known followees list. This effectively gives us a precision measure for the relevancy of our recommendations.

Figure 8(a) graphs the average precision versus recommendation list size, from the top-5 recommendations to the top-100 recommendations. Overall our recommendation strategy appears to perform well across the different recommendation-list sizes, generating precision scores of about 18% for a top-5 list size to an average of about 11% as the list size increases. Given the hard constraint that the target user is indeed a followee of a recommended user this should be viewed as a positive result, since we can also see that the precision does not seem to decline, but remains fairly constant with increasing recommendation-list sizes. It is also interesting to note that our results seem to be consistent with those reported in [19] for recommendation strategies where individual users were modeled by their own tweets as well as by the tweets of their followees.

An additional measure for evaluating recommendation performance is the position of relevant recommendations within the recommendation list, since users usually tend to focus their attention on items presented at the top of recommendation results. Therefore, a strategy, which consistently produces relevant recommendations in the top-half of the list, can be considered to be superior to a strategy which exhibits the same recommendation precision but presents relevant results in the bottom-half of the list. In **Figure 8(b)** we plot the average position of the relevant recommendations versus recommendation-list size. From this figure we can see that the average position of relevant recommendations ranges from approximately 3.5 (when k = 5) to well below 50 (when k = 100), a clear indicator that our proposed approach is able to position relevant recommendations towards the top-end of generated recommenddation-lists.

8. Conclusion

In this paper, we proposed a content based strategy in order to effectively model Twitter users by their tweets (and by the tweets of their followees), and utilized their social graphs in order to expand them and recommend potentially interesting users to follow. We performed an offline evaluation, based on a real Twitter user dataset, and the results obtained suggested that our content classification approach was able to provide accurate recommendations, despite the noisy nature of the training data. These encouraging results allowed us to consider further extensions of our approach into other application domains (and/or other information streams), so as to deepen our understanding of

Copyright © 2013 SciRes.

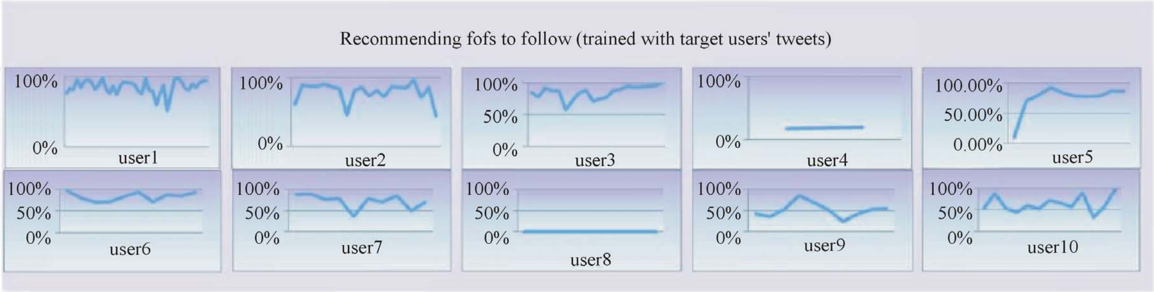


Figure 6. Recommending fofs to follow (trained with target users' tweets).

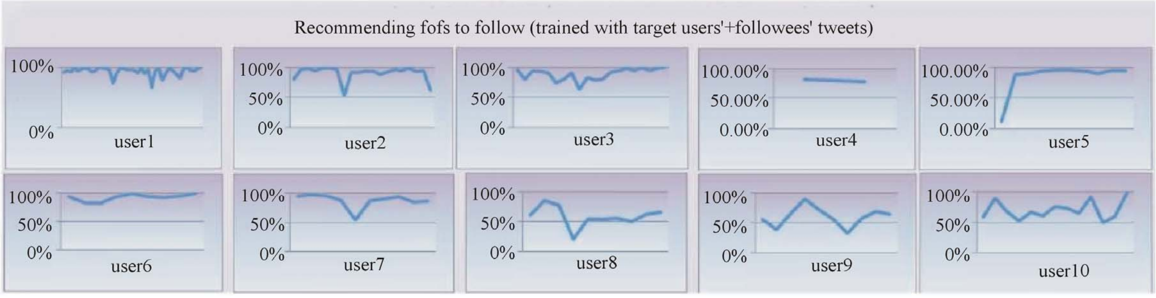
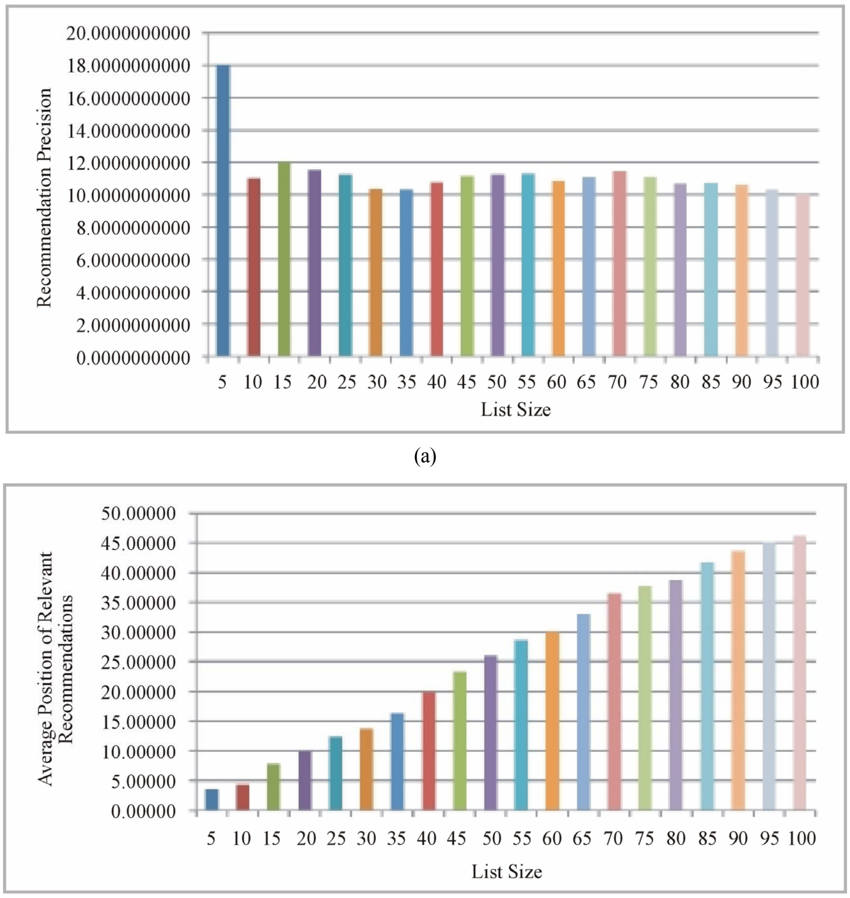


Figure 7. Recommending fofs to follow (trained with target users' + followees' tweets).



(a)

(b)

Figure 8. Average precision and average relevant recommendation position versus recommendation-list size. (a) Average precision; (b) Average position.

combining efficient methodologies for the design of high quality recommender systems.

REFERENCES

[1] B. Stuart and B. Martin, "Continuance Usage Intention in Microblogging Services: The Case of Twitter," *Proceedings of the* 17*th European Conference on Information Systems* (*ECIS*), Verona, 8-10 June 2009, pp. 556-567.

[2] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a Social Network or a News Media?" *WWW* '10: *Proceedings of the* 19*th International Conference on World wide web*, New York, 26-30 April 2010, pp. 591-600.

[3] S. Gaudin, "Twitter Now has 75M Users Most Asleep at the Mouse," 2010. http://www.cio.com/article/524515/Twitter_Now_has_75M_Users_Most_Asleep_At_the_Mouse

[4] W. S. Yerazunis, "Crm114 Revealed," 2006. http://crm114.sourceforge.net/docs/CRM114_Revealed_20061010.pdf

[5] "Seven Hypothesis about Spam Filtering," *TREC*, 2006.

[6] B. A. Huberman, D. M. Romero and F. Wu, "Social Networks That Matter: Twitter under the Microscope," Technical Representative, 2008. http://www.hpl.hp.com/research/scl/papers/twitter/

[7] C. Wagner and M. Strohmaier, "The Wisdom in Tweetonomies: Acquiring Latent Conceptual Structures," *Semantic Search Workshop at WWW*, 2010.

[8] F.-Y. Wang, K. M. Carley, D. Zeng and W. Mao, "Social Computing: From Social Informatics to Social Intelligence," *IEEE Intelligent Systems*, Vol. 22, 2007, pp. 79-83. http://dx.doi.org/10.1109/MIS.2007.41

[9] D. Gaffney, "#Iranelection: Quantifying Online Activism," Proceedings of the WebSci10: Extending the Frontiers of Society On-Line," Raleigh, 26-27 April 2010. http://journal.webscience.org/295/2/websci10 submission 6.pdf

[10] J. W. Owens, K. Lenz, and S. Speagle, "Trick or Tweet: How Usable Is Twitter for First-Time Users?" 2009. http://www.surl.org/usabilitynews/112/twitter.asp

[11] D. Tunkelang, "Tunkrank What Is Tunkrank," 2010. http://tunkrank.com/about

[12] M. Cha, H. Haddadi, F. Benevenuto and K. P. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy," *Proceedings of the* 4*th International AAAI Conference on Weblogs and Social Media*.

[13] D. D. Avello and D. J. Brenes, "Overcoming Spammers in Twitter—A Tale of Five Algorithms," *Conference on Information Retrieval*, 2010. http://www.slideshare.net/daniel.gayo/overcomingspammers-in-twitter-a-tale-of-five-algorithms

[14] J. Pazzani, M. J. Muramatsu and D. Billsus, "Syskill & Webert: Identifying Interesting Web Sites," *AAAI/IAAI*, Vol. 1, pp. 54-61.

[15] J. R. Mooney, and L. Roy, "Content-Based Book Recommending Using Learning for Text Categorization," *Proceedings of ACM DL*'00, pp. 195-204.

[16] J. Chen, W. Geyer, C. Dugan, M. Muller and I. Guy, "Make New Friends, but Keep the Old: Recommending People on Social Networking Sites," *Proceedings of the CHI*'09, 2009, pp. 201-210. http://dx.doi.org/10.1145/1518701.1518735

[17] W. Geyer, C. Dugan, D. R. Millen, M. Muller, and J. Freyne, "Recommending Topics for Self-Descriptions in Online User Profiles," *Proceedings of the 2008 ACM conference on Recommender systems*, New York, 2008, pp. 59-66. http://doi.acm.org/10.1145/1454008.1454019

[18] P. Melville, J. R. Mooney and R. Nagarajan, "Content-Boosted Collaborative Filtering," *Proceedings of* 2001 *SIGIR Workshop on Recommender Systems*, 2001.

[19] J. Hannon, M. Bennett and B. Smyth, "Recommending Twitter Users to Follow Using Content and Collaborative Filtering Approaches," *RecSys*, 2010, pp. 199-206.

[20] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. H. Chi, "Short and Tweet: Experiments on Recommending Content from Information Streams," *Proceedings of CHI*, 2010, pp. 1185-1194. http://www-users.cs.umn.edu/echi/papers/2010CHI/Zerozero88-tweet-recommender-ASC-PARC.pdf

[21] D. Tunklang, "A Twitter Analog to Pagerank", 2009, http://thenoisychannel.com/2009/01/13/a-twitter-analog-to-pagerank

[22] N. A. Christakis, "The Dynamics of Personal Influence," 2009, http://hbr.org/web/2009/hbr-list/dynamics-of-personalinfluence

[23] A. K. McCallum, "Bow: A Toolkit For Statistical Language Modeling, Text Retrieval, Classification And Clustering," 1996. http://www.cs.cmu.edu/mccallum/bow

[24] C. Siefkes, F. Assis, S. Chhabra and W. S. Yerazunis, "Combining Winnow and Orthogonal Sparse Bigrams for Incremental Spam Filtering," *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases*, Springer-Verlag, New York, Inc., New York, 2004, pp. 410-421. http://portal.acm.org/citation.cfm?id=1053072.1053110

[25] F. Assis, W. S. Yerazunis, C. Siefkes and S. Chhabra, "CRM114 versus Mr. X: CRM114 Notes for the TREC 2005 Spam Track," *Proceedings of* 14*th Text REtrieval Conference* (*TREC*), 2005.

[26] S. Chhabra, W. S. Yerazunis, and C. Siefkes, "Spam Filtering Using a Markov Random Field Model with Variable Weighting Schemas," 4*th IEEE International Conference on Data Mining*, 2004, pp. 347-350.

Copyright © 2013 SciRes.