Tadvise: A Twitter Assistant Based on Twitter Lists

Peyman Nasirifard and Conor Hayes

Digital Enterprise Research Institute
National University of Ireland, Galway
IDA Business Park, Lower Dangan, Galway, Ireland
firstname.lastname@deri.org

Abstract. Micro-blogging is yet another dynamic information channel where the user needs assistance to manage incoming and outgoing information streams. In this paper, we present our Twitter assistant called Tadvise that aims to help users to know their followers / communities better. Tadvise recommends well-connected topic-sensitive followers, who may act as hubs for broadcasting a tweet to a larger relevant audience. Each piece of advice given by Tadvise is supported by declarative explanations. Our evaluation shows that Tadvise helps users to know their followers better and also to find better hubs for propagating community-related tweets.

Keywords: Micro-blog, Twitter, People-Tag, Information Sharing

1 Introduction

In this paper we present Tadvise (`http://tadvise.net`), a novel application to assist Twitter users to select which followers would best be able to propagate the message to a relevant community-oriented audience. Tadvise automatically adds such well-connected hubs to a tweet to attract their attention. Hubs are considered as those followers, who have more well-connected topic-sensitive followers than others. Our approach is mainly based on Twitter lists. Twitter lists can be perceived as a way of *tagging* people [2]. Our work (Tadvise) uses Twitter lists for building user profiles in order to make recommendations on tweet diffusion. Tadvise is most useful for those Twitter users interested in sharing information, recommendations and news (such as conference announcements and events) with like-minded users in a community. Earlier work [8, 3] demonstrated the community (i.e., highly reciprocal network) structure of the Twitter network. As such, the scope of our work is focused on community-related pass-along tweets. For example, tweets like "deadline extended for next drupal conference..." are considered to be in the scope of Tadvise, as they are relevant to a particular interest group. On the other hand, informal status updates such as "having breakfast now..." are out of scope of Tadvise. We analyse the followers of a seed user (followers at *distance of 1*) plus the followers of the followers of the seed (followers at *distance of 2*) when considering the relevant audience for a (re)tweet. While not actually following the seed, followers at distance of 2 may be influenced by or be interested in a seed's community-related tweets, due to the dense community structure of the network [8, 3] and principle of locality [1]. Our focus is not to prohibit users generating and submitting novel contents, but to understand their followers' communities better.

2 Tadvise Overview and Components

Tadvise builds user profiles for twitterers in order to recommend tweets or retweets that could be potentially relevant to a community of their followers. To register for Tadvise, a twitterer u chooses to follow the Tadvise Twitter account (i.e., @Tadvise). Once notified, Tadvise crawls the social network of u and builds user profiles of her followers. After completing these steps, which are performed offline, Tadvise sends a direct message to u, indicating that it is ready to provide advice. By visiting the Tadvise homepage, u can benefit from advice and/or tweet a message directly to Twitter. Current version of Tadvise uses a traffic light metaphor to indicate its advice. A green light means that the majority of u's followers were tagged with one or more (hash)tags that exist in the tweet. The red light means that none of u's followers were tagged with the (hash)tags in the tweet. Finally, the amber light means that some of u's followers were tagged with the (hash)tags in the tweet, but they are not the majority of u's followers.

Tadvise has three main components, namely a *crawler*, a *user profile builder* and an *advice engine*. In the following, we describe all three components. Before proceeding any further, we formally define a Twitter-like system: A system S with n nodes (users) $U = \{u_1, u_2, ...u_n\}$, where there exists a set of unidirectional relationships R between users, so that if u_i makes a relationship ($r_{ij} \in R$) with u_j, we call u_i a follower of u_j and u_j a followee of u_i. We denote this relationship with $u_i \rightarrow u_j$. We assume that the system S is open, so that any user can make relationships with other users. The set of followees and followers of u_i are denoted by U_i^{fr} and U_i^{fo} respectively. User u_i can assign zero or more tags ($\{t_1, t_2, ...t_m\}$) to each of her followees. We define a function *lists* that gets a user u_j as input and returns pairs (u_i, t_k) meaning that u_i has assigned t_k to u_j.

2.1 Crawler of Tadvise

The crawling component of Tadvise gets a seed as input and uses the Twitter API for crawling twitterers. The crawling component does its job in two steps. First, it crawls the network of followers at distance of one and two of a seed (i.e., breath-first mechanism). The second step of crawling consists of crawling Twitter lists. This step takes the network of followers from the first step and crawls Twitter lists associated with each follower. Each API call returns 20 lists membership of a user. We put a limit (i.e., 300) on the number of Twitter lists associated with a user that we crawl, as 300 tags are reasonably enough for building a high-quality user profile for our purpose.

2.2 User Profile Builder of Tadvise

In order to assess the relevance of a tweet to a single user u_j, we create a weighted user profile for u_j containing metadata for u_j's communities, interests, expertise, etc. In short, each user profile is composed from metadata extracted from Twitter lists (tags) associated with the user by other users. In order to build a weighted user profile, we need to rank the tags that have been associated with a user (i.e., rank the result of $lists(u_i)$.) We do this by ranking the users who assigned the tags. There have been several studies of user ranking on Twitter [3, 8, 4] with no one technique demonstrating superiority. As such we make use of Kwak et al.'s finding [4] that a simple in-degree measure behaves similarly to PageRank on the Twitter network (see equation 1). As Twitter is an open platform and the connections are not necessarily reciprocal and does not require confirmation of the followee-side for public accounts, we do not consider the outgoing links (i.e., followees) for ranking purposes.

$$rank(u_i) = \log(\#U_i^{fo}) \tag{1}$$

Note that our ranking method can be generalised to a recursive one (see equation 2). In brief, users, who have more high-ranked followers, have higher ranks.

$$rank(u_i) = \sum_{u_j \in U_i^{fo}} rank(u_j) \tag{2}$$

$$weight(t_k, u_j) = \sum_{(u_i, t_k) \in lists(u_j)} rank(u_i) \tag{3}$$

The weight of a particular Twitter list for a target user profile is calculated by summing up the rank of people, who have assigned that Twitter list description to the target person (see equation 3).

As Twitter lists consist of arbitrary phrases, we use the Porter stemming algorithm [6] to reduce the number of unique terms. For tags that comprise more than one term, we use the stemmer on each term.

2.3 Advice Engine of Tadvise

The advice engine component takes user profiles and a tweet as inputs and provides two kinds of real-time diffusion advice: a) audience profiling that allows users to identify the subset of their followers that were tagged with a term used in the tweet; and b) recommending well-connected topic-sensitive users for a tweet, who may retweet the tweet.

Given a tweet and a user u_i, first we extract tags from the tweet. Typically, twitterers use the hashtags to specify particular topics (e.g., #drupal). We extract such tags from the tweet and enrich them using Google Sets (`http://labs.google.com/sets`). Enriching hashtags is important, as it may give us a set of tags that are semantically relevant to the original tags. Our analysis suggests that Google Sets provide more contextually relevant suggestions than lexical databases such as WordNet. Moreover, we also analyse the URLs within a tweet. Using regular expressions, we extract HTTP and FTP URLs from a tweet. Then we use the delicious API (`http://delicious.com/help/api`) to retrieve the tags associated with each URL. We do not enrich delicious tags, as delicious recommends already sufficient tags for a given URL. We then merge the tags from delicious and Google Sets.

For the first part of the diffusion advice (i.e., detecting the tags that are relevant to majority of the followers), we build aggregated user profiles that comprise user profiles of all followers of a seed at distance of 1 and 2 (i.e., summation). We represent such aggregated profiles as $followersProfile1(u_i)$ and $followersProfile2(u_i)$ respectively. These profiles contain (sorted) weights of all tags which were assigned to followers and also followers of the followers of a seed. Moreover, we cluster the sorted weights in $followersProfile1(u_i)$ and $followersProfile2(u_i)$ into two partitions which represent frequently occurring (thus highly weighted) lists and infrequently occurring lists. Rather than applying a fixed threshold to each profile, we find a *knee point* between the two partitions by applying the k-means clustering algorithm with k=2. The first partition, which groups high-ranked tags, represents the source of green light for the traffic light. The second partition represents the source of amber light advice. Tadvise shows the red light, if it is unable to find any representative tags of a tweet within either partition. Note that the traffic light metaphor was not aimed to prohibit users of generating novel contents.

Algorithm 1 shows pseudocode of the second part of the diffusion advice (i.e., recommending several well-connected topic-sensitive followers). The input of this algorithm is a directed graph g which is built as follows: The root of g is the seed u_i. We also add all members of U_i^{fo} to g ($u_j \rightarrow u_i$). The reason is that when u_i tweets a message, all of her followers receive that tweet and thus can act as potential hubs. Then, those followers of each follower of u_i, who were tagged with one or more (hash)tags in the tweet, will be added to g (using $followersProfile2(u_i)$). We pass g to the algorithm. The algorithm finds k hubs in g using In-degree so that the hubs cover as many interested followers (at distance of 2 of u_i) as possible and have as few overlapping followers as possible with each other. The reason that we also consider overlapping followers is to minimise redundant tweets, however, we envision allowing users to enable/disable this feature. The default value of k in the algorithm 1 is 3. The "hub score" in the algorithm 1 indicates the number of interested users, who potentially could receive a tweet through a hub. As tweets are 140-characters in length, we also consider the length of screen name of a hub, when making a recommendation. That means if two hubs disclose a tweet further with n users, we choose the hub, who has shorter length of screen name. We add the recommended candidates automatically to the tweet by inserting the screen name after the '@' sign and enable the user to tweet it directly from the Tadvise interface. In order to convince end users that our recommendations are relevant, we provide simple text-based explanations.

```
input  : Directed graph (g)
         Integer k // number of recommended hubs
output: candidates ⊂ g
candidates ← ∅;
covered ← ∅;
while size(candidates)!=k do
    calculate hubs in g and sort them based on hubs scores;
    node ← get the node with the highest score of hubs, so that
    followers(node) ∩ covered is minimum;
    candidates ← candidates ∪ node ;
    covered ← covered ∪ followers(node) ;
    g ← g − followers(node) − node ;
    if g == root(g) then  break;
end
return candidates;
```

Algorithm 1: Finding Well-Connected Hubs

3 Evaluation and User Study

We evaluated the following three main hypotheses. *Hypothesis 1*: Twitter lists assist twitterers to know each other better. *Hypothesis 2*: Users find it difficult to keep track of their followers. Tadvise helps users to know their followers (as a whole) better by identifying their communities, interests, expertise, etc. This hypothesis is important, because this may help users to boost communication and collaboration opportunities and may encourage users to tweet community-related tweets more often. *Hypothesis 3*: Tadvise helps users to propagate their community-related tweets more efficiently and effectively by proposing well-connected followers for a particular topic (instead of blind and ad-hoc retweeting requests.) The first two hypotheses are rather more general hypotheses and aimed to shed some light on (future) research on Twitter lists. The third hypothesis is the main one that is related to Tadvise functionalities.

3.1 Experiment - Design

In order to provide support for our hypotheses, we designed a survey that was personalised for each participant. For the survey design we studied the design recommendations of [7] and the well-known Questionnaire for User Interaction Satisfaction (QUIS) (`http://lap.umd.edu/quis/`). The survey had five main steps with a number of questions in each step. Most of questions in the survey had five possible replies: strongly agree, agree, neutral, disagree, and strongly disagree.

`Step 1: General Questions` - In the first step, the goal was to study: a) Whether subjects agree with the Twitter lists assigned to them; b) Whether the lists that were assigned to them fall into certain categories; and c) Whether the lists they assign(ed) to others fall into certain categories.

The aforementioned categories refer to common people-tagging categories discovered in a large-scale analysis of tagging behaviour [5]. They are as follows: *Characteristic* (e.g., friendly, cool), *Interest and Hobby*, *Affiliation* (e.g., IBM), *Working Group*, *Location*, *Name* (e.g., Peter, Mary), *Project*, *Role* (e.g., boss), *Skill and Expertise*, *Sport*, and *Technology* (e.g., drupal, semantic-web).

Steps 2-4 were presented in a game-like fashion with the subject having to guess or choose from a set of answers. Each step had 4 sub-steps.

`Step 2: Usefulness of Twitter Lists/People-Tags` - In step 2, we collected data on usefulness of Twitter lists. For the first three sub-steps of step 2, we picked one random follower, who had been assigned to at least three Twitter lists by any user and was also a followee of the subject. Then, we asked the subject to assign three Twitter lists to the follower. After clicking the submit button, we fetched the real Twitter lists assigned to the follower and asked the subject whether the result was useful in knowing the follower better. In sub-step 2.4, we focused on the community of the subject and asked the subject to guess three Twitter lists that fit the majority of her followers. After submitting the result, we showed our analysis result (i.e., all Twitter lists of first partition of the $followersProfile1(subject)$) to the subject and asked, if it helps to know the community of her followers better.

`Step 3: Knowledge of Followers` - Step 3 of the survey measured how well subjects know their followers. In each sub-step, we showed a random Twitter list (fetched from $followersProfile1(subject)$) to the subject and asked two questions: 1) Approximate percentage of the followers, who were assigned to that list. And 2) The followers (from twenty random followers), who were assigned to that Twitter list. In sub-steps 3.1 and 3.2, we picked a random Twitter list from the first partition of the $followersProfile1(subject)$ and ensured that at least 50% (if possible) of the 20 random followers are correct answers. In sub-steps 3.3 and 3.4, we picked a random Twitter list from the second partition. We enabled the subjects to skip a Twitter list (maximum three times in each sub-step), if they could not understand its meaning. In order to prevent the subjects selecting all followers, we put a maximum limit on the number of followers that could be selected. After submitting the result, we showed correct percentages and the missing followers from the list and asked the subjects whether this information helped in knowing their followers/communities better.

`Step 4: Usefulness of Recommendations` - In step 4, we investigated whether subjects found Tadvise recommendations to be useful. In sub-steps 4.1 and 4.2, we showed a random Twitter list (as a topic) from the first partition of the $followersProfile1(subject)$ and asked the subject to select two well-connected followers who could propagate a tweet about the topic to a broader audience. We enabled the subjects to select two followers from drop-down boxes, each containing twenty random followers, two of which were the correct answer. For the sub-steps 4.3 and 4.4, we carried out the same experiment, but with the Twitter lists from the second partition. After submitting the result, we presented the subject with our recommended hubs and provided explanations to justify our recommendations. Subjects were asked whether they were sufficiently convinced to use the recommendations.

`Step 5: General Questions` - In the final step, we asked subjects several general questions. Among others, we asked the subjects if they would find it useful to receive advice on whether their followers may be interested in a particular tweet. We also asked the subjects if they would find it useful to receive advice about the most effective and well-connected hubs.

3.2 Experiment - Result

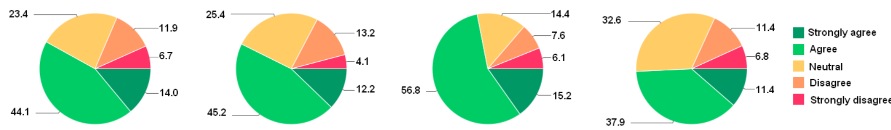
Participants Overview We made personalised online surveys for 112 Twitter candidates, among them 11 candidates did not fulfill our requirements for the survey - Each subject had to have at least three followers that been assigned to at least three Twitter lists, and who were also followees of the subject (i.e., reciprocal link). The survey was online for four weeks and we asked all 101 eligible candidates via email, instant messaging or direct tweet to participate in our survey. In total, 76 eligible candidates participated in our survey, among them 66 participants completed the survey. 47% of participants, who completed the survey (i.e., 31 participants) had 100 or more followers, among them twelve participants had more than 500 followers. Four participants had 1000 or more followers.

Results The results show that 79.1% of participants who were assigned to one or more Twitter lists mentioned that Twitter lists represent them correctly. Only 1.6% of participants claimed that they were assigned incorrectly to a list. Whether assigning lists or being assigned to lists, participants indicated that 96% of lists came from the following categories: Affiliation: 24.3%, Technology: 14.6%, Interest and Hobby: 15.9%, Skill and Expertise: 13.8%, Working Group: 9.2%, Location: 8.4%, Characteristic: 6.3%, Project: 3.8%, Role: 1.7%, Name: 1.3%, and Sport: 0.8%.

We used the results of sub-steps 2.1, 2.2, 2.3, 3.3 and 3.4 for evaluating our first hypothesis; sub-steps 2.4, 3.1, and 3.2 for evaluating our second hypothesis; and sub-steps 4.1, 4.2, 4.3, and 4.4 for evaluating our third hypothesis. Figure 1(a)-1(d) show the result for our hypotheses (refer to figures for the results). In step 5, 48.4% of participants were positive about being advised, if a tweet is relevant for majority of community-related followers, whereas 28.1% of participants were negative. The rest (23.5%) selected the *Undecided* option. 78.1% of participants were positive about being recommended hubs that could efficiently retweet a tweet and only 7.8% of participants found it useless. The rest (14.1%) selected the *Undecided* option.

4 Conclusion

In this paper we presented Tadvise, a system for helping users to manage the flow of messages in a micro-blogging network. We described our method for



(a) (b) (c) (d)

Fig. 1. Figure (a) is related to our first hypothesis: 58.1% of participants agreed that Twitter lists assist them to know their followers better, whereas 18.6% disagreed; figure (b) is related to our second hypothesis: 57.4% of participants agreed that Tadvise helps them to know their followers/community better, whereas 17.3% disagreed; figures (c) and (d) are related to our third hypothesis: 72% of participants found Tadvise recommendations and explanations for propagating *community-related* tweets convincing, whereas 13.7% disagreed (figure (c)); moreover, 49.3% of participants found Tadvise recommendations and explanations for propagating *non-community-related* tweets convincing, whereas 18.2% disagreed (figure (d)).

profiling the followers in a user's network and for giving advice on whom are well-connected topic-sensitive hubs in relation to a tweet. The result of our personalised evaluation surveys suggests that participants were mainly interested in being recommended hubs that can effectively retweet their messages and they found Tadvise recommendations for (mainly) community-related tweets useful and convincing.

Acknowledgments This work is partially supported by Science Foundation Ireland (SFI) under Grant No. SFI/08/CE/I1380 (Lion-2 project).

References

1. Chen, J., Nairn, R., Nelson, L., Bernstein, M., Chi, E.: Short and tweet: experiments on recommending content from information streams. In: CHI '10. pp. 1185–1194. ACM (2010)
2. Farrell, S., Lau, T., Nusser, S., Wilcox, E., Muller, M.: Socially augmenting employee profiles with people-tagging. In: UIST '07. pp. 91–100. ACM (2007)
3. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: WebKDD/SNA-KDD '07: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis. pp. 56–65. ACM (2007)
4. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: WWW '10. pp. 591–600. ACM (2010)
5. Muller, M.J., Ehrlich, K., Farrell, S.: Social tagging and self-tagging for impression management. Tech. rep., IBM Watson Research Center (2007)
6. Porter, M.F.: An algorithm for suffix stripping. Program 14(3), 130–137 (1980)
7. Shneiderman, B., Plaisant, C.: Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition). Pearson Addison Wesley (2004)
8. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: WSDM '10. pp. 261–270. ACM (2010)