Recommending Twitter Users to Follow Using Content and Collaborative Filtering Approaches

John Hannon, Mike Bennett, Barry Smyth
CLARITY Centre for Sensor Web Technologies
School Of Computer Science & Informatics
University College Dublin
{firstname.lastname}@ucd.ie

ABSTRACT

Recently the world of the web has become more *social* and more *real-time*. Facebook and Twitter are perhaps *the* exemplars of a new generation of social, real-time web services and we believe these types of service provide a fertile ground for recommender systems research. In this paper we focus on one of the key features of the social web, namely the creation of relationships between users. Like recent research, we view this as an important recommendation problem — for a given user, U_T which other users might be recommended as *followers/followees* — but unlike other researchers we attempt to harness the real-time web as the basis for profiling and recommendation. To this end we evaluate a range of different profiling and recommendation strategies, based on a large dataset of Twitter users and their tweets, to demonstrate the potential for effective and efficient followee recommendation.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; H.3.5 [**Online Information Services**]: Web-based services

General Terms

Algorithms, Experimentation

Keywords

Web 2.0, Twitter, Collaborative Filtering, Content Based Recommendation

1. INTRODUCTION

Few would have predicted the impact of *social media* on the Web, and yet today, blogging, reviews, wikis, and tagging are as much part of the Web as HTML and javascript. Fewer still would have dared to predict the success of Twitter, surely the poster-child of Web 2.0. Twitter is simplicity

itself. Borrowing ideas from social networking and SMS messaging, it allows users to post 140-character text messages (or *tweets*) to a constantly updating *public timeline* of user messages. Users can *tune* in to tweets by explicitly *following* other users, and so access a filtered timeline of messages from these people. The simplicity of Twitter is one of its most powerful features, and what started out as a simple way for users to provide status updates to their followers has grown to accommodate a wide variety of use-cases, from political campaigning to education, and from emergency news reporting to marketing and public relations. For example, Twitter was used extensively throughout the 2008 US Presidential Campaign[1] with Barack Obama successfully incorporating Twitter as part of a broader social media campaigning platform with which to co-ordinate and inform his followers and supporters. During the 2008 Mumbai attacks eyewitnesses used Twitter to post tens of tweets per second about the unfolding tragedy[2]. Twitter users on the ground helped create a list of the dead and injured, while others posted vital information such as emergency numbers, hospital locations for blood donations, etc. All of this information flowed at a pace that was far beyond what traditional forms of communication and news broadcasts could hope to generate. Today Twitter is used by many as a form of RSS reader, as users follow their favourite bloggers and news organisations. It has also proven to be a very popular way of sharing pages, causing some commentators to speculate about the potential for social media services like Twitter to represent a significant threat to the major search engines as the means by which users discover new content. And of course the advertisers and marketeers have also recognised the potential of Twitter, as a way to engage with customers in real-time.

In this paper we consider Twitter from a user modeling and recommendation viewpoint. We are motivated by Twitter's potential as a powerful source of profiling data. This is a novel take on profiling and recommendation in itself. For example, up until now most profiling and recommendation approaches have assumed the availability of high-quality interest and preference information, such as user ratings, purchase histories, or other forms of transaction logs. The real-time web, and Twitter in particular, seems far more limiting from a profiling and recommendation viewpoint. After all, user tweets are limited to only 140 characters in length and user's tweet on a wide variety of topics, often in a shorthand

---

[1] http://www.wired.com/politics/onlinerights/magazine/17-02/ff_obama
[2] http://www.wired.com/dangerroom/2008/11/first-hand-acco

that is likely to be opaque to conventional natural language processing techniques. Nevertheless, the sheer volume of real-time data that is available on Twitter makes for a tantalising profiling proposition and it is in this context that we seek to explore what might be feasible in practice.

To this end we describe a number of different profiling strategies, in order to learn about the interests of individual users, and we demonstrate how this information can be used to help people to benefit more from Twitter. To get the most from Twitter users must carefully select other people to follow, so that they can benefit from their tweets; the people a particular user, U, chooses to follow will be called $U's$ *followees* to distinguish them from the people who follow U, or $U's$ *followers*. With this in mind we develop a *followee recommender* for Twitter; see also [2]. We describe a recommender system that is designed to harness Twitter's real-time information as a source of recommendation knowledge for followee recommendation. We explore a variety of different recommendation strategies including content-based techniques [6, 13] which rely on the content of tweets, and collaborative filtering style approaches, based on the followees and followers of users, as well as a number of hybrid strategies. The technical details of our *Twittomender* system (http://twittomender.ucd.ie), and the profiling and recommendation algorithms that we use are discussed in detail. We then go on to describe two separate and complementary evaluations of these techniques and the Twittomender system. In the first we adopt a standard off-line recommendation evaluation using real-world data collected from some 20,000 Twitter users in order to compare the relative accuracy of our core recommendation techniques. Following this, we summarise the results of a recent open, live-user trial of the Twittomender system itself. The results of each evaluation speak to the potential for recommendation techniques to play a significant role in Twitter and on the real-time web more generally.

2. TWITTOMENDER: RECOMMENDING USERS ON THE REAL-TIME WEB

Twitter currently offers only very rudimentary search services to help people find new users to follow. This represents a significant opportunity for recommender systems. The basic assumption of this work is that the Twitter activity of a user's social graph (their tweets, and the tweets of their followers and followees) provides a powerful source of profile information that can be used as the basis for recommendation. In this section we describe our *Twittomender* recommender system, focusing on the system architecture, how users are profiled, and how these profiles can be used to suggest interesting users to follow.

2.1 System Architecture

The Twittomender system has been developed as a Web service; see Figure 1. It uses the Twitter API [3] to build and maintain a database of Twitter users, their tweets, their followers and their followees. Each user is represented by a variety of different sources of information, related to their recent Twitter activity and their social graph, which we shall discuss along with an in depth look at the key components of the system architecture in the following subsections.
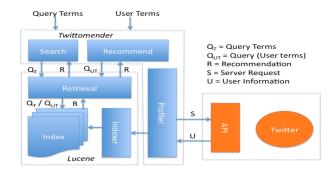
---

[3] http://apiwiki.twitter.com

Figure 1: The Twittomender System Architecture.

The system provides users with access to two basic modes of operation, as follows:

1. *User Search* — In this mode the user provides query-terms to receive a ranked-list of relevant Twitter users. Figure 2 shows the result of a search for people relevant to the query "social search". Each result is a Twitter user that has been indexed by Twittomender, alongside relevant information such as the user's username, description, popular terms from their recent tweets, and their most recent tweets. The searcher can click on the username to view their Twitter history or they can chose to follow the user directly.

2. *User Recommendation* — In this mode the user's own Twitter profile acts as a form of query to generate proactive recommendations of users to follow. In Figure 3 we see the recommendation-list generated for a particular Twitter user. The query box shows the terms extracted from the user's profile, as a query, and the result-list is a set of users who are judged to be relevant to the target user. To use Twittomender, users must *sync* their existing Twitter account with Twittomender so that a suitable profile can be generated.

Figure 2: An example of query-based search.

In what follows we will focus how users are represented and profiled in Twittomender and how this information is harnessed across a variety of different recommendation strategies, some of which rely on the content of tweets themselves, while others leverage the users' social graphs. We will return to the Twittomender system towards the end of this paper

Figure 3: An example of the recommendation interface.

when we describe the results of a recent live-user trial of the system in action.

2.2 Profiling Users on Twitter

The search and recommendation functionality provided by the Twittomender system is based on the availability of user profiles that reflect the likely interests of users, at least in terms of their Twitter histories. In this work we will look at some different sources of tweet information available for profiling. To begin with we consider the simplest source of profiling information, the user's own recent tweets. Thus, as per Equation 1, for a *target user*, U_T, let $tweets(U_T)$ be the set of recent tweets for U_T; in this work we will assume that $tweets(U_T)$ is the user's 100 most recent tweets. In this way $tweets(U_T)$ provides the basis for a content-based approach to user profiling, obviously under the assumption that users are likely to tweet about things that interest them.

$$tweets(U_T) = \{t_1, ..., t_k\} \quad (1)$$

This can be extended further. Each Twitter user *follows* a set of other users, their *followees*, and each user is *followed by* a set of users called their *followers*; see Equations 2 and 3. We can reasonably assume that the tweets of their followees and followers may provide further insights into a user's interests.

$$followees(U_T) = \{f_1, ..., f_m\} \quad (2)$$

$$followers(U_T) = \{g_1, ..., g_n\} \quad (3)$$

User's actively select their followees, probably because they expect their tweets will be of interest, and thus we can use their tweets in much the same way as a user's own tweets, as a complementary source of profile information. Thus, $followeetweets(U_T)$ is the set of tweets of the followees of U_T (see Equation 4).

$$followeetweets(U_T) = \bigcup_{\forall f_i \in followees(U_T)} (tweets(f_i)) \quad (4)$$

$$followertweets(U_T) = \bigcup_{\forall g_i \in followers(U_T)} (tweets(g_i)) \quad (5)$$

In turn, the follower's of U_T make the active decision to follow U_T, presumably because they (U_T's followers) expect U_T's tweets to be of interest. But will the tweets of these

follower's be of interest to U_T? And can they be used as a viable source of profiling information as per Equation 5? Neither questions can be affirmed with confidence yet, since, in the case of the majority of Twitter users at least, users exert little control over their followers; users rarely prune away followers that don't interest them and there are many cases of followers who generate very few tweets themselves. Nevertheless, the tweets of followers certainly provides an intriguing source of profile information worth exploring.

In summary the above suggests 5 basic profiling strategies: (1) representing users by their own tweets ($tweets(U_T)$); (2) by the tweets of their followees ($followeetweets(U_T)$); (3), by the tweets of their followers ($followertweets(U_T)$); (4) by the ids of their followees ($followees(U_T)$); (5) or by the ids of their followers ($followers(U_T)$). And of course, as we shall discuss, these different *sources* of profile data can be combined in various ways, so that, for example, we could represent a user by their own tweets, the tweets of their followees, and the tweets of their followers.

2.3 Indexing & Recommendation

Now that we have a basis for profiling Twitter users, based on tweets and/or social connections, we can index these profiles and develop the recommendation framework to deliver results based on a target user profile, or indeed a specific set of query terms. We have chosen to develop this framework using the open source Lucene platform[4]. There are numerous advantages to proceeding in this fashion, as opposed to developing a bespoke recommendation framework. For a start, Lucene provides a proven, robust, and scalable indexing and retrieval platform that is designed to cope with Web-scale data and usage. In addition, it provides access to powerful indexing and term-weighting features that will accommodate a more sophisticated approach to user profiling than a simple frequency-based term-weighting scheme. Finally, Lucene's retrieval functions can be used directly for the query-based retrieval of profiles and can be readily adapted for recommendation.

Since Lucene is a text-based search engine, its basic units of information are documents to be indexed and stored for retrieval. We can treat profiles as documents, which, after all, are simply collections of words (from tweets) or other user ids. Thus, using Lucene's indexing features we can represent each, U_T, as a weighted term-vector, $profile(U_T, source)$ (see Equation 6), such that the ith element of this vector represents the ith unique term in *source*, and the weight of this i^{th} term (w_i) represents the importance of this term for U_T. In the case where *source* is one of the content sources ($tweets(U_T), followeetweets(U_T)$ or $followertweets(U_T)$) then these terms will be the words used in the tweets of the relevant users, whereas when *source* is one of the social/collaborative sources ($followees(U_T)$ or $followers(U_T)$) then these terms will be user ids. In what follows we will use $profile(U_T)$ instead of $profile(U_T, source)$ without loss of generality in cases where the *source* parameter is clear.

$$profile(U_T, source) = \{w_1, ..., w_n\} \quad (6)$$

We could use a simple frequency count as the term weighting function, so that the profile vector could be made up of the frequency counts of the various words used in the tweets of U_T, for example. However, in this instance we use

[4]http://lucene.apache.org/java/docs/

Lucene's TF-IDF weighting metric; thus, the TF-IDF score of term t_i in U_T is proportional to its frequency of occurrence in $profile(U_T)$ and inversely proportional to its frequency across the other profiles, U, as shown in Equation 7 to 9. This results in a higher weighting for profile terms that are frequent in a given profile but infrequent across the profile-base as a whole, which helps to distinguish profiles during retrieval by discounting matches on common terms (words or users). For example, if we represent each user just by their own raw tweets, then the TF-IDF weighting will give a higher weight to terms that are common to U_T but unusual across the rest of the user population. These high-scoring terms serve to better *distinguish* U_T's interests relative to the other users, during retrieval.

$$TF - IDF(t_i, U_T, U) = tf(t_i, U_T) \bullet idf(t_i, U) \qquad (7)$$

$$tf(t_i, U_T) = \frac{n_{i,T}}{\sum_k n_{k,T}} \qquad (8)$$

$$idf(t_i, U) = log \frac{|U|}{|\{d : t_i \epsilon d\}|} \qquad (9)$$

Query-based retrieval and profile-based recommendation are then implemented using Lucene's standard retrieval function, with the target user's *profile document* serving as the search query in the case of the latter. All of this provides a very powerful and flexible retrieval and recommendation framework, since profiles can be represented and indexed by a combination of source terms, effectively harnessing a variety of different recommendation strategies, from simple content-based or social recommendation strategies to more sophisticated hybrids. For instance by using the content sources we can generate a space of content-based recommender systems. Conversely, by indexing users just by their followees or followers or both we can generate collaborative filtering style [11, 16] recommenders.

3. OFF-LINE EVALUATION

The success of our recommender will ultimately depend on its ability to suggest new users who are likely to be worth *following*, by the target user; remember, we are interested in recommending *followees* as opposed to *followers*. In this section we describe an offline evaluation of a variety of different recommendation techniques based on a comprehensive dataset generated from real Twitter users. This particular approach to evaluation is commonplace amongst recommender systems research and allows us to compare the recommendation effectiveness of a variety of different profiling and recommendation strategies. On its own however it only tells part of the story and in the next section we complement this evaluation with a live-user study of Twittomender.

3.1 Data & Algorithms

Twittomender is designed to incrementally extend its profile database each time a new user allows Twittomender to connect to their Twitter account. Once deployed, we hope that the system will prove popular enough to attract significant user numbers. However, for the purpose of this research we needed access to a critical mass of existing users. To do this we imported 20,000 users directly using the Twitter API. We began with a small seed-set of 15 users (basically the followees and colleagues within our research group) and

Table 1: Evaluation Datasets.

Users	Tweets	Words	Followers	followees
1000	80	15	664	321
19,000	78	14	465	520

expanded the user-base by following their *follower* and *followees* links. For each user we also downloaded up to 100 recent tweets. We split the data-set into two sets of users – one containing 1000 users to act as *test users*, and a larger *training-set* of 19,000 users; see Table 1 for a summary of these data-sets in terms of their median tweets, followees, and followers per user and the median words per tweet per user. In this experiment we evaluate 9 different profiling and recommendation strategies based on the different sources of profile information, in isolation and in combination. To begin with we implemented four *content-based* strategies that rely on the content of tweets as follows:

1. (*S1*) users are represented by their own tweets ($tweets(U_T)$);

2. (*S2*) users are represented by the tweets of their followees ($followeestweets(U_T)$);

3. (*S3*) users are represented by the tweets of their followers ($followerstweets(U_T)$);

4. (*S4*) a hybrid strategy in which users are represented by the combination of tweets from $tweets(U_T)$, $followeestweets(U_T)$, and $followerstweet(U_T)$;

In addition we implemented three *collaborative filtering* style strategies, in the sense that we view a user profile as a simple set of user ids.

5. (*S5*) users are represented by the IDs of their followees ($followee(U_T)$);

6. (*S6*) users are represented by the IDs of their followers ($follower(U_T)$);

7. (*S7*) a hybrid strategy in which users are represented by the combination $followee(U_T)$ and $follower(U_T)$;

Concretely, for each of the above profiling strategies we constructed a separate Lucene index and then used the standard Lucene retrieval engine to generate recommendations. Finally we implemented two hybrid, *ensemble* strategies, $S8$ and $S9$. In each case the ensemble is composed by a selection of previous basic component recommenders, $S1 - S7$ and the union of the recommendations from these independent strategies is scored and ranked. $S8$ and $S9$ vary in terms of the scoring function that they use as follows:

8. (*S8*) the scoring function is based on a combination of content and collaborative strategies $S1$ and $S6$;

9. (*S9*) the scoring function is based on the position of the user in each of the recommendation lists so that users that are frequently present in high positions are preferred over users that are recommended less frequent or in lower positions.
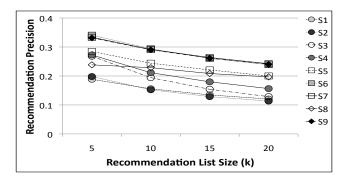
Figure 4: Average precision vs. recommendation-list size for the 9 different recommendation strategies using the 1000-user test sets.

Figure 5: Average relevant recommendation position vs. recommendation-list size for the 9 different strategies using the 1000-user test sets.

During the evaluation we tested these 9 algorithms on users in the 1000-user test-set. For each of these target users we generate a query by taking the *top-20* highest scoring terms from the user's profile; in other words we just use a portion of the users full profile information based on the TF-IDF weightings assigned to the profile terms/ids. This profile query is used to generate a recommendation-list for each of the 9 strategies. For each target profile we count how many of the recommendations are in the user's known *followees* list. We call these *relevant recommendations*; in other words, we look to see how often the recommender suggests people that the target user is known to have followed. In the sections that follow we will describe the summary results across these different algorithms, focusing on the average overlap between recommendation lists and followees-lists, and the position of these relevant recommendations. And we do this for different recommendation list sizes (k) from the top-5 recommendations to the top-20 recommendations.

3.2 Recommendation Precision

Our basic measure of recommendation performance is the average percentage overlap between a given recommendation list and the target user's actual followees-list; this is effectively a *precision* measure. To begin with, Figure 4 graphs the average precision versus recommendation-list size for the 9 different recommendation strategies using the 1000-user test-sets. Overall the different recommendation strategies appear to perform well across the different recommendation-list sizes, generating precision scores, in the 1000-user test-set, of between 11% (for strategy S2 at $k = 20$)and almost 25% (for strategy S7), for example. This should be viewed as very positive results since the success metric here — namely, that the target user is a followee of a recommended user — can be viewed as setting a reasonably high relevance standard; in Twitter, becoming a followee of a user is a deliberative act and most users limit who they follow to avoid being swamped with irrelevant messages. We can also see that relevant recommendations tend to be clustered towards the top of recommendation lists since the precision of all strategies is seen to decline within increasing recommendation-list size. Interestingly, the collaborative strategies perform better than the content strategies; $S5 - S7$ consistently outperform $S1 - S4$ across all values of k. We also find that strategy $S2$ tends to perform poorly. For example, for recommendation lists of size 10, strategy $S2$ delivers a precision score of only 0.15; so only 1 or 2 of these 10 recommendations
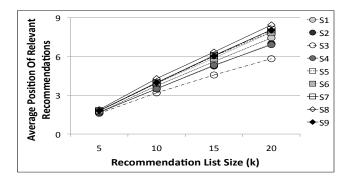
are actually followees of the target user. In contrast, we find the tweets of a user's followers (that is the people who follow the target user), $S3$ to perform significantly better. For instance, for recommendation lists of size 10, strategy $S3$ delivers a precision score of 0.2; a 33% relative increase in precision over $S1$. This suggests that the tweets of your followees are not necessarily a good predictor of these same followees. The two ensemble hybrids ($S8$ and $S9$) perform well with $S9$ delivering precision values that are as good as the best individual strategy, although it is notable that it does not do any better than the best individual strategy, $S6$.

3.3 Ranking Effectiveness

Of course the percentage of overlapping recommendations is just one measure for evaluating recommendation performance. The position of relevant recommendations is also an important consideration, especially since we know that users focus the lion's share of their attention on items at the top of results or recommendation-lists. Thus, two strategies may perform well in terms of their overall precision, but if one consistently produces relevant recommendations in the top-half of the list, while the relevant recommendations for the other tend to appear in the bottom-half of the list, then, all other things being equal, the former strategy can be considered to be superior.

In Figure 5 we plot the average position of the relevant recommendations versus recommendation-list size for the 9 recommendation strategies. In general the strategies perform similarly across different recommendation-list sizes — the average position of relevant recommendations ranges from approximately 1.7 (when $k = 5$) to just over 7 (when $k = 20$) — and it should be clear that all of the strategies are capable of positioning relevant recommendations towards the top-end of their recommendation-lists.

Interestingly, the general trend seems to be that strategies which do less well from a precision viewpoint seem to do better from a position viewpoint, especially in larger recommendation-lists. For example, in the above precision results, the collaborative strategies out-performed the content-based strategies, but the reverse is true in the case of these position results; for example, for a recommendation-list of size 20 the typical content strategy recommends relevant users at about position 7, compared to position 8 for a typical collaborative (or hybrid) strategy.
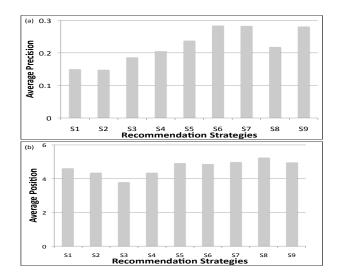
Figure 6: Summary precision (a) and position (b) results.

Table 2: Twittomender Trial Participants.

Stats	Total
Number Of Participants	34
Median Number Of Followees	66
Median Number Of Followers	71
Median Number Of Tweets	273

Figure 7: The number of relevant recommendations for different recommendation-list position ranges.

3.4 Summary Results

So in summary there is reason to be optimistic about the potential for a followee-recommender for Twitter using the techniques described here. In each case, we have found our content-based and collaborative filtering profiling approaches to be capable of delivering recommendation-lists that include a reasonably high percentage of relevant users appearing towards the top of these lists. There is somewhat of a trade-off between recommendation-list precision and the average position of relevant recommendations, and the various strategies are influenced by the size of the recommendation-list. By way of a summary, Figure 6 (a & b) present the *mean average precision* and the *mean average position* results for the above experiments; in each case we average the precision and position values over the different values of k.

4. A LIVE-USER TRIAL

It is worth returning to the manner in which we evaluate precision in the previous offline experiments. Precision is calculated as the percentage overlap between recommendations and the target user's existing followees-list, but it would be unwise to discount the non-overlapping recommendations as definitively not relevant to the target user. They are 'not relevant' only in the sense that they are not already followees of the target user, and it would be incorrect to assume that these recommendations are not of interest to the target user. They may indeed be of great interest to the user. As such we view these results as providing a useful baseline with respect to likely recommendation precision in a live-user context, and in this section we describe just such a trial based on the usage patterns of 34 trial participants during March 2010. These participants were all existing Twitter users. Summary information is shown in Table 2.

The Twittomender System is available online[5] and for the purpose of this trial we invited interested users to synchronise their Twitter accounts with Twittomender to in-

clude their tweets and social graphs in the Twittomender database. For the purpose of this trial, Twittomender was configured to use profiling and recommendation strategy $S9$, meaning that a combination of all of the different sources of profiling information was brought to bear on recommendation. Upon synchronising their Twitter account with Twittomender (so that their profiles could be generated from their tweets, followees, and followers) each participant was presented with a list of 30 recommended Twitter users (from a Twittomender database of approximately 100,000 users at the time of this trial) and the user was asked to indicate which of the recommended users they would likely follow. There are two important points to make here: (1) none of the users existing followees or followers were included in these recommendation-lists – they were filtered out pre-recommendation; (2) participants understood that for the purpose of this trial their indicating that they would likely follow a given user would not in anyway effect their live-Twitter social graph (in other words they wouldn't actually end up following the user in question).
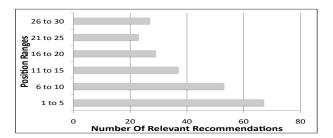
On average, the 34 participants indicated a willingness to follow an average of 6.9 users per recommendation-list and the majority of these relevant recommendations appeared towards the top of the recommendation-lists. For example, Figure 7 shows a histogram of the number of relevant recommendations for different recommendation-list position ranges and we can see that 120 out of a total of 236 relevant recommendations appeared in the top 10 recommended users. We view this result to be very positive. Every participant found at least some new users worth following in their recommendation-list and an average of almost 7 new users to follow per recommendation-list would certainly help to drive 'followee relationships' in the Twitter universe.

During this trial we also provided users with an opportunity to test the query-based search functionality of Twittomender. Very simply, users were encouraged to enter a standard query with a view to receiving recommendations for users who are likely to be relevant to this query. Out of the 34 participants, 31 tested the search service. They

---

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

Figure 8: A Wordle tag cloud of the search query terms submitted to Twittomender.
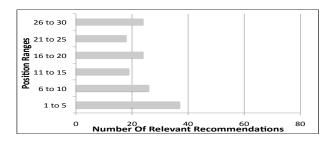

Figure 9: The number of relevant search results for different recommendation-list position ranges.

submitted queries with an average of 3.7 terms on a variety of topics and once again received recommendation-lists containing 30 users; for convenience we have summarised their query terms in the form of a Wordle tag cloud (wordle.com) in Figure 8. We tracked how often a user received a search result representing a user they would be inclined to follow; once again we filtered-out any users any existing followees and followers from these result-lists. This time users indicated they would be willing to follow an average of 4.9 of the suggested users, per search. Obviously this indicates a lower number of relevant users, compared to the 6.9 relevant users per recommendation-list for the profile-based recommendations, but this is probably to be expected since the short search queries provided by users carry far less (albeit perhaps more focused) information than the richer profile queries used during recommendation. Nevertheless, we view an average of 5 new users worth following per search to be again a very strong result from a Twitter perspective. In Figure 9 we show a histogram of the number of relevant results for different recommendation-list position ranges. We see a relatively even spread of relevant users across the top 30 results, with a bias towards the top ranking results.

5. RELATED WORK

Our research is motivated by the availability of real-time web data and the potential use of this unique form of user-generated content to drive recommender systems. To this end it is useful to consider 3 related areas of research: (1) understanding the form and function of the real-time web; (2) the role of user-generated content in recommendation; and (3) our application area of people recommendation.

There is currently considerable research attention being paid to Twitter and the real-time web in general. RTW services provide access to new types of information and the real-time nature of these data streams provide many opportunities and challenges. Moreover, companies like Twit-

ter and Yahoo have opted to make their data available and Twitter's developer API provides researchers with access to a huge volume of information for example. It is no surprise then that the recent literature includes a number of interesting analysis of Twitter's real-time data, largely with a view to developing an early understanding of why and how people are using services like Twitter; see for example [8–10,12]. For instance, the work of Kwak et al [12] describes a very comprehensive analysis of Twitter users and Twitter usage, covering almost 42m users, nearly 1.5bn social connections, and over 100m tweets. In this work the authors have examined reciprocity and homophily among Twitter users, they have compared a number of different ways to evaluate user influence, as well as investigating how information diffuses through the Twitter ecosystem as a result of social relationships and retweeting behaviour.

The research presented in this paper is obviously not focused on analysing microblogging usage per se. Rather our interest is piqued by the potential to use this near-ubiquitous form of user-generated content as a source of preference and profiling information in order to drive recommendation. User-generated content is inherently noisy but it is plentiful, and recently researchers have started to consider its utility in recommendation. For example, there has been some recent work [14] on the role of tags in recommender systems, and recently researchers have started to leverage user-generated reviews as a way to recommend and filter products and services. For instance, [1] look at the use of user-generated movie reviews from IMDb as part of a movie recommender system and similar ideas are discussed in [17]. Both of these examples of related work look to mine review content as an additional source of recommendation knowledge, but they rely on the availability of detailed item reviews, which may run to hundreds of words, which may not always be the case. In this paper, we are interested in the more challenging form of user-generated content that comes from micro-blogging services like Twitter and we wish to use Twitter content as a way to drive automated recommender systems that are capable of responding to the preferences of users.

Finally, there has been considerable research over the past number of years to help users find and connect with people online [4, 5, 7, 15]. For example the work of Guy et al. [6] has looked at the use of recommender systems to identify people that you might wish to invite into your social network, focusing on an enterprise context. In this work the researchers explored profiling users across a number of different sources of information in enterprise so that explicit relationships could be highlighted based on the fact that two users contributed in similar ways to similar information sources; for example users may share patent authorships or they may be closely related according to the organisation hierarchy within the enterprise, or they may co-author papers together or contribute to the same wikis. This information can then be used to identify similar users and proactively make recommendations to users as a way to drive relationship creation within social media. The results of the live-user trial confirmed that this type of contextualized followee recommendation was capable of driving a significant uplift in the formation of new relationships within the enterprise. Related work, at least in terms of its core motivation to drive relationship building, has been carried out by Freyne et al [4] and Geyer et al [5] who have explored a number of recommendation techniques for improving user engagement

within social media and social networks. We see the work of this paper fitting neatly in to this body of research — our interest in recommending followees is largely driven by the need to help Twitter users to create interesting connections with remote Twitter users. However, our work differs in some important ways. In particular, we focus on noisy, unstructured micro-blogging data, which is far from the more structured data (co-authorship links, community membership data etc.) that others have used in similar endeavours. A central and, we believe, novel contribution of this work is to demonstrate that, noisy as Twitter data is, it can still provide a useful recommendation signal. Perhaps closest to the work in this paper is very recent work by Chen et al [3] on the development and evaluation URL recommendation strategies to demonstrate the utility of using various combinations of tweet content and social graph information during recommendation. Like the present work, Chen et al are clarifying the significant role that real-time web data can play in future recommender systems research.

6. CONCLUSION

In this paper we propose that Twitter users can be usefully modeled by the tweets and relationships of their Twitter social graphs. We have demonstrated how these profiles can be used as the basis for a followee recommender called Twittomender. An offline evaluation, based on real-user data, suggests that this recommender system is capable of delivering high-quality recommendations, and recent results from a live-user trial of the Twittomender system supports this finding. We see the core contribution of this paper as two-fold. On the one hand we have demonstrated the potential of the real-time web, and micro-blogging services like Twitter, to serve as a useful source of recommendation information. Our analysis suggests that noisy as Twitter content is, it does provide a useful profiling and recommendation signal. We have shown how existing recommendation strategies can be usefully harnessed to solve important challenges — followee recommendation in this case — in the real-time web.

Acknowledgment

7. REFERENCES

[1] S. Aciar, D. Zhang, S. Simoff, and J. Debenham. Recommender system based on consumer product reviews. In *WI '06: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 719–723, Washington, DC, USA, 2006. IEEE Computer Society.

[2] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy. Make new friends, but keep the old: recommending people on social networking sites. In *Proceedings of the 27th international conference on Human factors in computing systems*, pages 201–210, New York, NY, USA, 2009. ACM.

[3] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. H. Chi. Short and tweet: experiments on recommending content from information streams. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, pages 1185–1194, 2010.

[4] J. Freyne, M. Jacovi, I. Guy, and W. Geyer. Increasing engagement through early recommender intervention. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 85–92, 2009.

[5] W. Geyer, C. Dugan, D. R. Millen, M. Muller, and J. Freyne. Recommending topics for self-descriptions in online user profiles. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 59–66, New York, NY, USA, 2008. ACM.

[6] I. Guy, I. Ronen, and E. Wilcox. Do you know?: recommending people to invite into your social network. In *IUI '09: Proceedings of the 13th international conference on Intelligent user interfaces*, pages 77–86, New York, NY, USA, 2009. ACM.

[7] W. Hsu, A. King, M. Paradesi, T. Pydimarri, and T. Weninger. Collaborative and structural recommendation of friends using weblog-based social network analysis. In *AAAI Spring Symposium Series*, 2006.

[8] B. A. Huberman, D. M. Romero, and F. Wu. Social networks that matter: Twitter under the microscope. *SSRN eLibrary*, 2008.

[9] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Procedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop*, pages 56–65, 2007.

[10] C. Jensen, J. Davis, and S. Farnham. Finding others online: reputation systems for social online spaces. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 447–454, New York, NY, USA, 2002. ACM.

[11] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. Grouplens: applying collaborative filtering to usenet news. *Commun. ACM*, 40(3):77–87, 1997.

[12] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW '10: Proceedings of the 19th international conference on World Wide Web*, pages 591–600, 2010.

[13] M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The Adaptive Web*, pages 325–341. Springer Verlag, 2007.

[14] S. Sen, J. Vig, and J. Riedl. Tagommenders: connecting users to items through tags. In *WWW '09: Proceedings of the 18th international conference on World Wide Web*, pages 671–680, New York, NY, USA, 2009. ACM.

[15] A. Seth and J. Zhang. A social network based approach to personalized recommendation of participatory media content. In *ICWSM '08: Proceedings of the 2nd international conference on Weblogs and Social Media*, 2008.

[16] U. Shardanand and P. Maes. Social information filtering: algorithms for automating "word of mouth". In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[17] R. T. A. Wietsma and F. Ricci. Product reviews in mobile decision aid systems. In *PERMID*, pages 15–18, Munich, Germany, 2005.