Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media

Personalized Recommendation of Twitter Lists Using Content and Network Information

Vineeth Rakesh, Dilpreet Singh, Bhanukiran Vinzamuri, Chandan K. Reddy
Department of Computer Science, Wayne State University, Detroit, MI
{vineethrakesh, dilpreet, bhanukiranv, creddy} @wayne.edu

Abstract

Lists in social networks have become popular tools to organize content. This paper proposes a novel framework for recommending *lists* to users by combining several features that jointly capture their personal interests. Our contribution is of two-fold. First, we develop a `ListRec` model that leverages the dynamically varying tweet content, the network of *twitterers* and the popularity of lists to collectively model the users' preference towards social lists. Second, we use the topical interests of users, and the list network structure to develop a novel network-based model called the LIST-PAGERANK. We use this model to recommend auxiliary lists that are more popular than the lists that are currently subscribed by the users. We evaluate our `ListRec` model using the Twitter dataset consisting of 2988 direct list subscriptions. Using automatic evaluation technique, we compare the performance of the `ListRec` model with different baseline methods and other competing approaches and show that our model delivers better precision in terms of the prediction of the subscribed lists of the *twitterers*. Furthermore, we also demonstrate the importance of combining different weighting schemes and their effect on capturing users' interest towards Twitter lists. To evaluate the LIST-PAGERANK model, we employ a user-study based evaluation to show that the model is effective in recommending auxiliary lists that are more authoritative than the lists subscribed by the users.

Keywords: Twitter, lists, recommendation.

1 Introduction

Social content recommendation has risen to a new dimension with the advent of microblogging platforms like Twitter, FriendFeed, Dailybooth, and Tumblr. As the number of people using such platforms are increasing on a daily basis, there is a rapid growth in the amount of data and information gathered using such microblogs. Although, this uproar of data provides us with a "gold-mine" of real-world information, it is not without it's side effects; it has lead to a major problem called *the information overload* (Borgs et al. 2010). The most critical problem that branches out from the information overload is the difficulty in organizing the timeline of users. For example, an active *twitterer* follows 80 users on an average, and receives over 1000 tweets (Qu and Liu 2011); due to such an incessant flooding of user-timeline, many important and interesting tweets remain unnoticed by the users. Furthermore, this results in an increase of irrelevant and personal tweets that are not worth reading. Researchers have tackled the problem of information overload from various different perspectives such as organizing trending topics in user's timeline, URL recommendations for *twitterers*, recommending followers and tweets (Bernstein et al. 2010; Abel et al. 2011; Armentano, Godoy, and Amandi 2012; Hannon, Bennett, and Smyth 2010). A new direction of research that is proposed in this paper is the development of personalized recommendation based on *social lists*. Lists serve a dual purpose in various social networks. First, they serve as a newsletter or a daily-digest for users who seek unified source of information. Second, they act as topical-hubs that unite users who share similar interests. Originally lists were introduced by Twitter in 2009; however, they have been adopted by various social networking websites in different forms under different names. For instance, Google+ terms lists as *social circles* and Facebook provides a feature called *community pages*. In general, every list has a curator who creates the list and makes it as private or public. Other users can freely subscribe to such public lists, while private lists are restricted to the owner's approval. Lists are one of the strongest indicators of topical homophily (Kang and Lerman 2012). Consequently, they can be an excellent tool to smoothen the problem of information overload.

Recommending lists is a challenging task because most users create them for grouping friends or other users whom they find interesting. Such lists that are created for personal convenience do not gain the attention of people. This implies that most of them do not have any subscribers. Furthermore, list names are not unique; there can be thousands of lists with similar (or even same) names (Kim, Jo, and Moon 2010). This further exacerbates the problem of finding genuine, authoritative and topically relevant set of lists.

In this paper, we propose two recommendation models that recommend lists for Twitter users based on their personalized interest. Our first model, called the `ListRec`, captures and models the users' interest based on a combination of content, network and trendiness based measures. For users with rich tweet history, we measure their interests using the topics derived from their tweets. Unlike the existing studies, we view the *twitterer's* interest as a temporally varying feature and exploit this variation using an exhaustive set of streaming tweets to dynamically model the users' interest. For users with sparse tweet history, we project the user space into a followee space and utilize the followee's list subscriptions to indirectly measure the interest of the users. We also add a new trend based score that measures the popularity of lists in the Twitter domain. The final score is then modeled as a linear combination of these three individual scores (based on content, network, and popularity) to effectively measure the interests of the users and personalize list recommendation. The coefficients in this linear combination are estimated using a cyclic ridge regression estimation approach. Our experimental results show that the `ListRec` outperforms other competing state of the art methods. Our second model is the LIST-PAGERANK which will recommend lists that are popular and are more (topically) authoritative than the lists that are currently subscribed by the users. To the best of our knowledge, there are no studies that use Twitter lists for personalized recommendation. We summarize the major contributions of this paper as follows:

a. We propose a recommendation framework called `ListRec` that recommends Twitter lists based on the personalized interest of *twitterers*. Unlike the existing studies that recommend external information like news articles and blogs, our work is purely domain-specific.

b. The interests of users are modeled using a combination of weighting schemes: (a) a content based scheme that models the users' interest based on temporally varying topics; (b) a network based scheme that uses the followee-network of the users to overcome the tweet sparsity; and (c) a trendiness based scheme that is based on the popularity of the lists.

c. We propose a LIST-PAGERANK based algorithm that leverages the network structure of Twitter lists to recommend authoritative lists that match the topical interest of the users.

The rest of this paper is organized as follows. We begin by describing the modeling of `ListRec` in Section 2. Section 3 describes the creation of the list network and formulation of the LIST-PAGERANK. Section 4 will show the results of our experiments and explain the data collection methodology. Section 5 discusses the related work on this topic. Finally, the conclusions obtained through this study are presented in Section 6.

2 User Interest Modeling

Researchers have proposed many variants of user interest modeling in Twitter. This includes the simple $Tf * Idf$ based weighting to more complex methods based on collaborative filtering, Co-Factorization Machines and concept graph on Wikipedia articles (Chen et al. 2012; Pennacchiotti et al. 2012; Hong, Doumith, and Davison 2013; Lu, Lam, and Zhang 2012). For the purpose of this study, we classify the Twitter users into two categories: *the persistent twitterers*, and the *active consumers*. *Persistent twitterers* are users who tweet frequently and consistently. Therefore, they tend to have a rich tweet history. On the other hand, *active consumers* are characterized by a sparse tweet history, but they actively consume information from Twitter by following other users. Our aim is to develop a list recommender system that can be effective for both these categories of users. For this reason, we use a combination of users' tweet history (when available), and their network of followees to collectively measure their personalized interest.

List-preference based on varying topical interests: The topical interest of *twitterers* changes with time. For example, consider the following set of tweets tweeted by a *twitterer* over a period of 1 year.

1. Love my #iphone 4s and its retina screen simply colorful and vibrant. #iphoneRocks - March 2012
2. #Apple versus #Samsung this is interesting. I think #iphone has lost it's charm. - December 2012
3. Finally sold my #iphone4s and got a #GalaxyS4 simply loving the big screen!. Can't wait to explore the new #Android - June 2013

We can clearly see the transition of the user's interest from iphone to Galaxy S4 mobile. This also means that recommending lists related to iphone might not be interesting to this user. Therefore, we model the interest of *twitterers* as a temporally varying factor by using the discrete dynamic topic model (dDTM) (Blei and Lafferty 2006) to create a *temporal topic-preference matrix* that captures the inclination of the users towards a set of topics at different time frames. Unlike LDA (Blei, Ng, and Jordan 2003), the dDTM sees the order of collection as an evolving set of topics. The dDTM uses a state space model on the natural parameters of the multinomial distributions that represent the topics. The alignment among topics across time steps is captured by a Kalman filter. The inferior performance of topic models over short text documents is a well known problem that has been widely studied in the literature (Hong and Davison 2010). To overcome this problem, we use tweet pooling technique (Mehrotra et al. 2013) to collect all the tweets tweeted by these users, and use their history of tweets as input to the dDTM. For the set of users $\mathcal{U}$ in our database, we run the dDTM over their tweet history to obtain the set of topics $\mathcal{T}$ at different time frames t_f. We then use these topics as an intermediate plane to formulate a *content-list* matrix that maps the topical interests of the *twitterers* to the set of lists L. We explain this mapping using the following set of matrices:

- *User-topic matrix J:* The topical interest of *twitterers* J for a time frame t_f is $|\mathcal{U}| \times |\mathcal{T}|$ matrix, where the value $\mathcal{UT}_{ij}$ denotes the number of times a word in *twitterer* u's tweet has been assigned to the topic $\tau_j \epsilon \mathcal{T}$.
- *Topic-List matrix M:* The topic-List matrix defines a relation between the set of lists and the topics that are spanned by these lists. We create this matrix by collecting the set of tweets that emerge from every list $l \epsilon L$, and use the dDTM to generate a set of topics. The topic-List matrix is represented as $M = |\mathcal{T}| \times |L|$.

The interest of *twitterers* towards the lists is a $|\mathcal{U}| \times |L|$ matrix that is obtained as follows:

$$\phi = J \cdot M \tag{1}$$

Network based List-preference: For users with low tweeting frequency (i.e. the active consumers), we use their followee network to indirectly measure the preference of user $u \epsilon \mathcal{U}$ to a set of lists $\{l_1, ..., l_n\}$ in L. First, we obtain the set of followees F for users in $\mathcal{U}$ to create a *user-followee* matrix given by

$$E = |\mathcal{U}| \times |F| \tag{2}$$

Second, the user's interest towards his followees is measured based on the number of times a user u_i retweeted his followee f_j. The adjacency matrix E is defined as follows:

$$E_{ij} = \frac{RT(i,j)}{\sum\limits_{f \in F} RT(i,f)} \tag{3}$$

where $RT(i,j)$ is the number of times the user i retweeted his followee j, and $\sum RT(i,f)$ is the total number of retweets by the user i (normalization factor).

Third, the list subscriptions of followees in the set F is retrieved to create a *followee-list* matrix $|F| \times |L|$ given by

$$J_{ij} = \begin{cases} 1 \text{ if } i \text{ subsribes to list } j \\ 0 \text{ otherwise} \end{cases} \tag{4}$$

Finally, we obtain the *network-list* matrix $|\mathcal{U}| \times |L|$ as follows:

$$\Delta = E \cdot J \tag{5}$$

List-preference based on Trending List: A list can be considered *trending* in Twitter if the hashtags produced by this list are popular at a specific time t. Therefore, for every list in the set L, we retrieve the hashtags that emerge from their respective tweets to create a hashtag-List matrix given by

$$K = |H| \times |L| \tag{6}$$

We then determine the *trending lists* by estimating the popularity of hashtags in the set H at a specific time t in the ordered Twitter streams $\mathcal{D}_1, \ldots \mathcal{D}_n$. Each Twitter stream $\mathcal{D}_i$ is a set of ordered n-tuples represented as $\{< h_{i1}, t_{i1} >, \ldots, < h_{im}, t_{im} >\}$ where h_i is the hashtag and t_i is it's corresponding publishing time. Kwak et al. (Kwak et al. 2010) showed that the topics in Twitter become popular for a certain period of time and gradually die. This encourages us to use a time-decay function to estimate the trending hashtags (Manjhi et al. 2005). The numeric weight of hashtags H in the Twitter stream D at any given time is a function of the elapsed time since the first occurrence of this hashtag. The common way to model such functions is using an exponential-decay. Mathematically, we denote the function as follows:

$$W(h) = \sum_{<h_i, t_i> \epsilon \mathcal{D}} \beta^{\frac{[t_{now} - t_i]}{T}} \tag{7}$$

where the parameter $\beta \epsilon (0,1]$ controls the weight of the hashtags; t_{now} denotes the current time, and T sets the granularity of time-sensitivity. In this paper, we give equal importance to every hashtag at the beginning by setting β to 1. The trendiness of hashtags $h \, \epsilon \, H$ is measured by estimating $\hat{W}(h)$ using first order derivative of their cumulative counts. The trending-List matrix is given by

$$\Omega = \mathcal{H} \cdot K \tag{8}$$

where $\mathcal{H}$ is a row matrix that contains the estimated weights for the hashtags H.

Recommendation score of Twitter List: The recommendation of Twitter lists L for a set of users $\{u_1, ..., u_n\}$ given their tweet history Z_u and followees F_u is represented as a linear combination of their topic based weightage ϕ, their network based score Δ, and the score based on list trendiness Ω. Formally, we denote the preference score by

$$P(u,l) = \alpha\phi + \beta\Delta + \gamma\Omega \tag{9}$$

Regression for List Recommendation In this section, we describe the algorithm to estimate α, β and γ for the preference score (9). The ridge regression algorithm is used as a solver for estimation at each step of Algorithm 1. The regularization function used here is the L_2 norm of the regression coefficient vector. We now explain the major steps involved in this estimation algorithm.

In the first step, β and γ are initialized using a fitting heuristic. In this fitting heuristic, we estimate β^{init} by selecting a randomly sampled subset of data, and fitting it to the response vector Δ. The size of this sample is set of 30% of the original data. Similarly γ^{init} is also estimated using this 30% fitting heuristic.

Algorithm 1 Cyclic Approximate Ridge Regression for List Recommendation

Require: Binary response vector P, Topic List matrix F, Network List Matrix Δ, Trending List Matrix Ω

1: Initialize β=β^{init} and γ=γ^{init}
2: $P^{'} = P - \Delta\beta^{init} - \Omega\gamma^{init}$
3: Using $P^{'}$ and F estimate α^{final}
4: Set $P^{'} = P - F\alpha^{final} - \Delta\beta^{init}$
5: Using $P^{'}$ and Ω estimate γ^{final} as in Step 3
6: Set $P^{'} = P - F\alpha^{final} - \Omega\gamma^{final}$
7: Using $P^{'}$ and Δ estimate β^{final} as in Step 3
8: Output α^{final},β^{final} and γ^{final}.

The values of β^{init} and γ^{init} are used in Equation (9) to formulate the regression problem to estimate α^{final}. This cycle of estimation is continued in the remaining steps to estimate the values of β^{final} and γ^{final} as explained in Algorithm 1. This final set of coefficient values are used for estimating the scores.

3 LIST-PAGERANK

The main goal of our LIST-PAGERANK model is to recommend auxiliary set of lists that are authoritative and topically similar to the lists that are subscribed by the *twitterers*. We begin this section by explaining the construction of the list network. We define the set of Twitter lists as a tuple L_c <C,M,J,S>, where C denotes the curator of the list;

M is the set of list members; J is the set of topical words, and S is the set of subscribers. A directed graph $D(V, E)$ is formed with lists as the vertex V of the network. Defining edges in Twitter lists can be tricky. This is because, unlike user-follower relationship in Twitter, an explicit relationship between lists does not exist. Therefore, in this paper, we exploit the hidden structure of Twitter lists to define their linkage. We say that, an edge between two lists exists *if the member of a list is a subscriber of another list*. Figure 1 shows this notion using three list nodes. In this figure, the user C who is a member of list 3 subscribes to another list 2; thus, establishing a linkage between these lists. Similarly, user G, a member of list 2 subscribes to list 3.

With the list network defined, we now explain the meaning of authority in Twitter list. The definition of authority is based on the following observations:

- *Influential twitterers tend to be a member of many lists.*
- *Lists containing influential twitterers have the potential to attract many subscribers. The subscription count in turn makes the list authoritative.*

This notion is similar to the real-life event of news paper subscription. Top circulated news papers like *The Wall Street Journal* and *The New York Times* attract more subscribers because the content produced by them are relevant and exhaustive; more importantly, they are written by prominent reporters and journalists. The influence of a *twitterer* can be measured by his list membership count. For example, the user C in Figure 1 is a member of two lists: list 1 and list 3. Now, the authority score of the list 2 goes up due to the presence of this influential member.

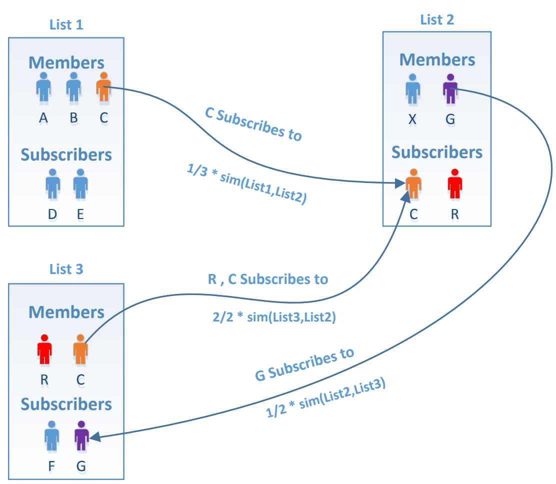

Figure 1: Representation of list-network using a subscriber-member relationship

We show this effect in Figure 2 by plotting the membership count of users against the subscriber count of their list subscription. We see that, as the membership count increases, the subscriber count also increases. The increase in subscriber count becomes more pronounced when the membership count goes beyond 80. This clearly shows that the subscription of users with high list membership results in

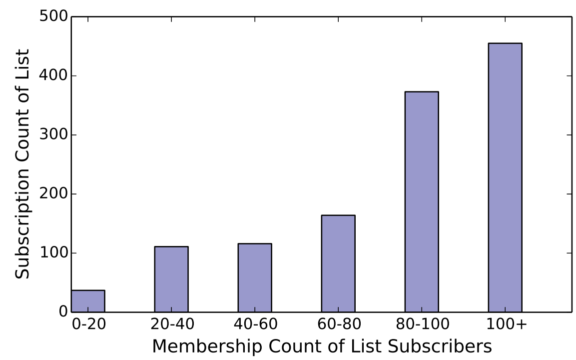

Figure 2: Influence of *twitterers'* membership count over their list subscription count

attracting more subscribers; thereby, making the list more dominant. Our goal is to recommend auxiliary set of lists that are not only authoritative, but also topically similar to the lists that are subscribed by the *twitterers*. The methodology of collecting the auxiliary list set L_c will be explained in Section 4.3. We now explain the creation of *list-topic* matrix J and the formulation of our LIST-PAGERANK.

To create the *list-topic* matrix J, we obtain the topics from the auxiliary set of lists L_c by tokenizing the tweets from each list. We then construct a bag-of-words vector W, where $W =< tf(l, w_1), ..., tf(l, w_m) >$ and $tf(l, w)$ is the *term frequency* of the tweet word w in the list $l \epsilon L_c$. The matrix J is denoted by $|L_c| \times |W|$. We define the adjacency matrix for our list-network as follows:

$$LN_{i,j} = \begin{cases} \dfrac{|M(i) \cap S(j)|}{|M(i)|} \times Sim(i,j) & \textit{if link exists} \\ 0 & \textit{otherwise} \end{cases} \tag{10}$$

In the above equation, the *link exists* if atleast one member of list i is a subscriber of the list j, $M(i)$ is the set of all members of i, and $S(j)$ is the set of all subscribers of j. The numerator $|M(i) \cap S(j)|$ denotes the number of members of list i who are subscribers of list j, and the denominator $|M(i)|$ is the total number of members of the list i. An example of this formulation is shown in the Figure 1. In this example, there is just one user (user C), who is both a member of list 1 and a subscriber of list 2. Therefore, the edge weight between list 1 and list 2 is $1/3$. Finally, the similarity term $Sim(i, j)$ is calculated as the cosine similarity between the lists i and j given by

$$Sim(i,j) = \frac{J_i \cdot J_j}{|J_i| \times |J_j|} \tag{11}$$

where J_i and J_j are topic vectors obtained from the *list-topic* matrix J.

In our list-network, it is possible for lists to form loops. For example, in the Figure 1, a loop exists between list 3 and 2 since the member of list 3 is a subscriber of list 2 and vice versa. Such loops will accumulate high influence without distributing it. In the *random surfer* algorithm of PageRank, a link is added from every web page to all other web pages to overcome this problem. We adopt the same methodology in our list graph, by introducing the teleportation vector LW defined as:

$$LW = J_k \tag{12}$$

where J_k is the k-th column of the *list-topic* matrix J. In this manner, the teleportation probability is higher for a list which is more topically similar to the original list.

We finally represent the LIST-PAGERANK as a convex combination of the matrix $LN_{i,j}$ and the *teleportation* vector LW as follows:

$$L_{rank} = \alpha LN_{i,j} + (1 - \alpha) * LW \tag{13}$$

The addition of the teleportation vector LW enables the surfer visiting a list to jump to another random list with a probability $(1 - \alpha)$, where α is a parameter that controls the probability of teleportation that is set between 0 and 1.

4 Experiments

4.1 Dataset Description

In the earlier section, we categorized the users as *persistent twitterers* and *active consumers* based on their tweeting behaviour. In general, it is difficult to obtain users with such characteristics merely by querying the Twitter for random user Ids due to the API limitations. For this reason, we use our streaming database that was collected from January 2012 to August 2013 using Twitter's *firehose* API that provides 10% of every day's streaming tweets. Figure 3 shows the comparison of the tweet frequency plots between users who appear in over 60% of our database, and users who appear in less than 30% of our database. While both the plots follow a powerlaw distribution, the former shows a tweet count between 500-1000 for a majority of users, while the latter clearly shows that most users have sparse number of tweets. We denote the set of users with high tweeting fre-

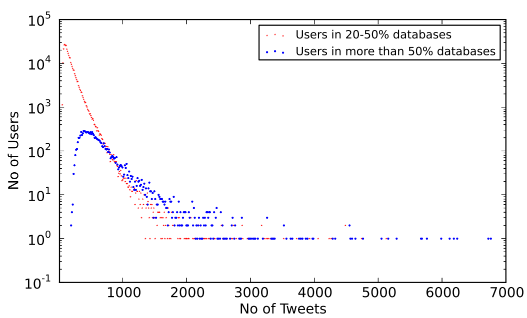

Figure 3: Tweeting frequency of frequent and infrequent *twitterers* from our streaming database

quency by P, and those with low frequency by $\mathcal{A}$. We create our user dataset $\mathcal{U}$ as follows:

1. The set of *persistent users* is denoted by $P^* = \{p | p \epsilon P$ and p has atleast 3 list subscriptions$\}$.
2. Since the *active consumers* have a scarce set of tweets, we choose these users based on their followee count. Figure 4 shows that most users in the set $\mathcal{A}$ do not follow other users. Therefore, we impose a threshold on the followee count of the users. Formally, we denote the set of *active consumers* by $\mathcal{A}^*$, where $\mathcal{A}^* = \{a | a \epsilon \mathcal{A}$ and a has atleast 10 followees and 3 list subscriptions $\}$.

Our final user dataset is given by $\mathcal{U} = P^* \cup \mathcal{A}^*$. For our experiments, we have $|P^*| = 529$, $|\mathcal{A}^*| = 221$ and $|\mathcal{U}| = 750$.

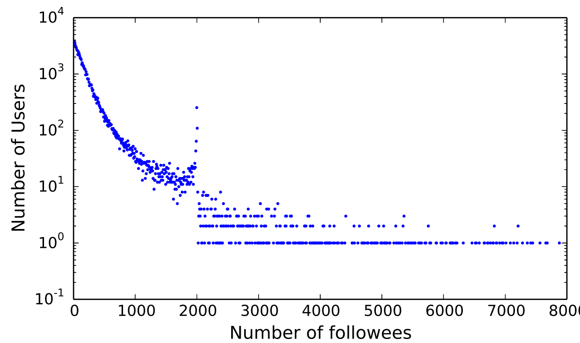

Figure 4: Number of followees per *active consumer*

4.2 Automatic Evaluation

We evaluate the `ListRec` model based on the assumption that a user who subscribes to a list finds it interesting. Our test dataset is the set of all users in $\mathcal{U}$, and their list subscriptions L, $|L| = 2988$. Ideally, the correct recommendation for a user $u \epsilon \mathcal{U}$ should correspond to the lists from his own direct subscription.

Evaluation Metrics For evaluating our model we use the standard information retrieval measures. For every user we compute: (1) *precision at rank k* (P@k) for our task is defined as the fraction of rankings in which the subscribed lists is ranked in the *top-k* positions, (2) Our recommendation is correct when the user-subscribed list is present in the ranked set of lists. The *mean reciprocal rank* (MRR) is the inverse of the position of the first correct list in the ranked set of lists produced by our model, and (3) *success at rank k* (S@k) is the probability of finding at least one correct list in the *top-k* ranked ones. (4) The *discounted cumulative gain* (DCG) (Järvelin and Kekäläinen 2002) is based on the simple idea that highly relevant lists are more important than marginally relevant lists. DCG computes the score for a list based on it's position in the ranked set of lists. It then calculates the cumulative gain by considering a linear summation of the relevance scores of the lists scaled by a logarithmic factor. The scaling helps in obtaining the discounted cumulative gain metric.

Method Comparison We compare the performance of our model to the following baselines

- `EntRank`: For every user, we collect the user entities mentioned in their tweets. The entity based ranking scheme ranks the lists based on the number of members who correspond to the entities mentioned in the user tweets.

Table 1: Comparison of the past and current interest of users generated by dDTM, and the topics generated by LDA without taking the temporal shift of user interests

User	List Topics	Past Interests (dDTM)	Current Interests (dDTM)	LDA Topics
A	Influencial in tech {Android, #technology, microsoft }	food, volunteer, tech	Google, #tech, hackathon	Volunteer, fastfood latish, meal
B	Top 50 funny {#smile, Darwin , follow}	media, poll, breaking	Comdey, Science, actors	people, critics media, news
C	Astronauts in space {NASA, #ISS, Mars }	Game, #redsox, mars	#mars, NASA, astronauts	Redsox, win space, game
D	US Senators {politics, #syria, Obama }	Bills, business, venture	governor, syria, policy	startup, venture legislation, pay
E	Marketing Industry {adcampaign, business, media}	kobe, payments, ipad	eComm, Advertising, Basketball	ipad, payments play, game

Table 2: Performance comparison between different methods using MRR and Precision metrics

Algorithm	MRR	P@1	P@5	P@10
`EntRank`	0.08	0.04	0.0418	0.032
`Trendiness`	0.006	0.0	0.0017	0.001
`Content`	0.48	0.40	0.29	0.23
`UserNet`	0.51	0.31	0.34	0.21
`listRec`*	0.36	0.32	0.33	0.21
`listRec`	**0.54**	**0.44**	**0.39**	**0.35**

Table 3: Performance comparison between different methods using Success at k and DCG metrics

Algorithm	S@5	S@10	S@30	DCG
`EntRank`	0.14	0.19	0.198	0.27
`Trendiness`	0.024	0.027	0.0307	0.15
`Content`	0.43	0.471	0.52	17.54
`UserNet`	0.427	0.432	0.481	18.2
`listRec`*	0.324	0.342	0.348	16.49
`listRec`	**0.45**	**0.493**	**0.54**	**18.42**

- `Trendiness`: We set $\alpha = \beta = 0$ to rank the lists purely based on trendiness.
- `Content`: The content based weighting scheme ranks the lists purely based on the topical interest of the Twitter users. We set $\beta = \gamma = 0$ for this scheme.
- `UserNet`: This scheme purely based on user-network. We set $\alpha = \gamma = 0$ for this scheme.
- `listRec`*: Instead of using dDTM to measure the user interest, we use the LDA by ignoring the temporal variation of topical interest.

Table 2 shows the results of MRR and precision, while Table 3 reports the results of success at k and DCG. We clearly see that the proposed `listRec` is the best performing model for all measures. The `Content` based scheme closely follows our model, this clearly emphasizes the fact that topical homophily is one of the important features. The `UserNet` performs reasonably well when compared to `listRec` and `Content`. This shows that the followee network of a *twitterer* plays an important role in determining his list subscription. In other words, the probability of a user subscribing to a list increases if the list has already been subscribed by his followees. This clearly shows the impact of the social circle on users' interest. The poor performance of `EntRank` indicates that the entities mentioned in the users tweet need not be the members of a list.

Finally, it is important to note that the `listRec`* performs poorly when compared to our `listRec` model. As mentioned before, the topic component of `listRec`* ignores the temporal variation of user's interest while generating the topics. From the results, it is quite conclusive that the poor performance of `EntRank` is due to absence of this temporal variation. We provide further insights on the performance of `listRec`* by comparing the topics generated by dDTM and LDA over users' tweets in Table 1. The topics generated by dDTM are split into two columns denoting the past, and the current interests of the *twitterer*. We can clearly see that there is a significant shift between the *twitterer's* past and current interests. For example, user A's past interest was related to topics like fastfood and meal, while his current interest is more towards technology related topics like Google, hackathon etc. Similarly, user C's past interest was mostly centered around games, while his current interest is inclined towards space related topics like NASA, mars, etc. The last column in Table 1 shows the topical interest of users generated using LDA. We can see that the topics are a mixture of the users' past and current interests, with a majority of topics emerging from user's past time frame. On the other hand, the topics from the users' list subscription have a greater match with their current interests rather than their past. This is the most important reason for the superior performance of `listRec` over `listRec`*. Figure 5 shows the DCG measure for the top 20 ranks. We can see that the `listRec` is able to suggest more related lists when compared to all other performance measures.

Table 4: Comparison of quality between subscribed and recommended lists for top topics

list Number	list Description	list Topic	Subscribed list			Recommended list		
			S-Cnt	Rt-Cnt	Entities	S-Cnt	Rt-Cnt	Entities
1	Technology News	#Apple, Technology	37	14	@magpienikki, @Shadowrayven	14092	104	@stevenbjohnson, @timoreilly
2	TV	#Glee, #NFL	4	2	@TVGuide, @TheEllenShow	1875	42	@RottenTomatoes, @davidschneider
3	Funny Tweets	Fun, Comedy	4	17	@FakeAPStylebook, @badbanana	8802	69	@OMGFacts, @GuyCodes
4	Politics	#Obama, Law	113	12	@ezraklien, @SFist	1145	114	@ABCPolitics, @foxnewspolitics
5	Aviation	InFligtCalls, #privatejets	32	15	@DBaviation, @bizjetkev	256	31	@flightglobal, @TheDEWLine

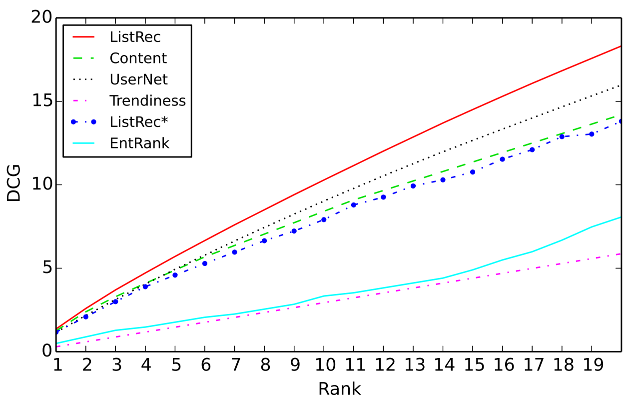

Figure 5: Average discounted cumulative gain of related lists for top 20 ranks for different algorithms

4.3 Empirical Evaluation

In this section, we compare the quality of lists that are subscribed by the users with the lists that are suggested by the LIST-PAGERANK. To run our experiments, we first create the list-network LN that was described in Section 3. Using the set of user-subscribed lists L as the seed set, we do the snow ball sampling using the following steps: (i) For every list $l \epsilon L$, we create a user-set by collecting all the members and the subscribers from l, and (ii) for every user in this user-set, we retrieve their list subscriptions. We iteratively perform these steps to create auxiliary set of lists L_c, $|L_c| = 10876$. The adjacency matrix LN is constructed using the set L_c.

Table 4 shows the comparison between the top 5 subscribed and recommended lists. We see that the recommended lists have extremely popular entities as members or subscribers. For instance, the list on *Technology News* have users like *stevenbjohnson* who is a popular media theorist; *timoreilly* is the CEO of O'Reilley Media. The list based on TV topics like #Glee, #NFL etc. have very famous actors like *davidschneider*, and movie rating website *RottenTomatoes* associated with them. Similarly, the lists on *Funny tweets* have very prominent entities associated with them and the lists on *Politics* have famous political news channels such as *foxnewspolitics* and *ABCPolitics*. The list related to aviation has *TheDEWLine* who is a prominent aerospace journalist and blogger and *flightglobal* which is a news source for global aviation. These entities are not only famous in real-world, but are also active users in Twitter domain. All these users have a large number of followers and retweeters. Additionally, the recommended lists have a large number of followers and retweets when compared to the subscribed lists. We can clearly see that the presence of prominent Twitter personalities acts as a magnet for attracting more subscribers and retweeters for a list.

Characteristics of the recommended list So far, we showed the performance of our LIST-PAGERANK model using qualitative comparisons between the subscribed and the recommended lists. We now show the characteristics of these recommended list by choosing samples from user-subscribed lists L using various criteria. To achieve this, we use the quality measure proposed by Weng et al. (Weng et al. 2010) for their TwitterRank model. However, unlike the authors, we don't use their method to evaluate our model; instead, we simply use it to measure the characteristics of the recommended list and compare it with the classical PageRank algorithm and the indegree measure. This is mainly because unlike the TwitterRank, the user is not a part of the list network. The sampling procedure for measuring the list characteristics is shown in Algorithm 2.

For all our experiments, we sample 20 lists from the user-subscribed list L. Therefore, we set $|P|$ as 20 in step 1 of Algorithm 2. The selection of the sample P is based on four different list-based features as described below:

Influence score of list members: Our first selection criteria is based on the influence score of the list members. We wanted to see whether there is any correlation between the authority of the lists that are calculated using individual authorities of the list member, and the authority of the list calculated by our LIST-PAGERANK model. To measure the influence scores of the list members, we use the popular *klout*

Algorithm 2 Sampling procedure for analyzing the list characteristics

Require: The user-subscribed set of lists L
1: Choose a sub-set $|P|$ from the set L using different list-based features
2: **for** each list $l \epsilon P$ **do**
3: Crawl a set of 10 auxiliary lists, denote this set as Z
4: Create a new list-network with the set Z
5: Run the LIST-PAGERANK algorithm to rank the lists in this new network Z
6: Using equation (14) report the characteristics of the ranked lists
7: **end for**

score [1] service that provides the influence score of twitter users using various inter and intra-domain based measures. For every list in L, we retrieve the members and calculate their klout scores. The klout score for a list is then calculated as the collective score of the individual members. To select the sample set P we rank the lists according to their klout scores and choose a set of lists P_{kh} (high klout score) from the 90th percentile and P_{kl} (low klout score) from the 10th percentile of the klout score counts, $P = P_{kh} \cup P_{kl}$.
Subscriber count of the list: In this criterion P is chosen based on the number of subscribers of the list. We calculate the subscriber count of each list in L, and rank them according to this count. We now choose the lists P_{sh} (high subscription count) from the 90th percentile, and the lists P_{sl} (low subscription) from the 10th percentile of the subscription count, $P = P_{sh} \cup P_{sl}$.
Retweet count of the list: Similar to the subscription count criterion, we rank the lists in L based on their retweet counts. We then choose lists P_{rh}, and P_{rl} from the 90th and 10th percentile of the retweet counts respectively.
Membership count of the list: The final criterion is based on the membership counts of the list. We rank the lists in L based on their membership counts. We then choose lists P_{mh} (high membership), and P_{ml} (low membership) from the 90th and 10th percentile of the membership counts respectively.

The characteristic score of the recommended list is measured using the following equation:

$$C(Z) = \{z_i | z_i \epsilon Z \text{ and } R(z_i) < R(z_p)\} \tag{14}$$

Where, z_p is the set of lists in Z which are directly subscribed by the users, and $R(z_i)$ denotes the rank of the list i. According to the equation (14), $C(Z)$ measures the number of auxiliary lists that have a higher recommendation score than the subscribed lists. A high score of $C(Z)$ implies that a major part of the recommended list is from the auxiliary list, while a low score implies most recommendation are from the user's direct list subscription.

We show the results of running our LIST-PAGERANK over the sample network Z in Figure 6. The x-axis denotes the different characteristic features that were used to choose

[1] http://klout.com/

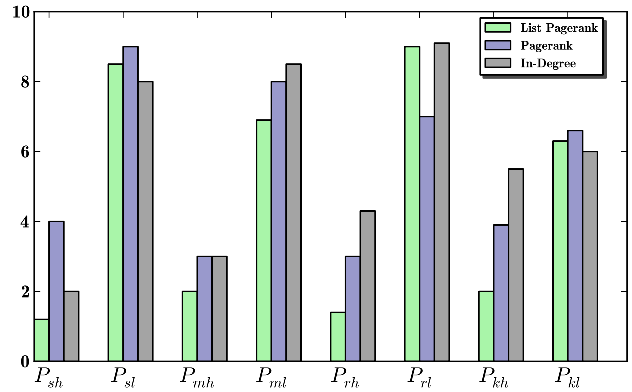

Figure 6: Characteristic score of ranked lists for different features

the user-subscribed list P, and the y-axis denotes the characteristic score obtained using the equation (14). From this figure, we can infer three important characteristics of the recommended lists. When we choose the set P with high subscription count (P_{sh}), the lists recommended by our model is mostly from the subscribed set of lists rather than the auxiliary (crawled) set of lists. This trend is similar in both the PageRank and In-degree algorithms; nonetheless, the PageRank and In-degree tend to recommend more auxiliary lists when compared to our model. If the user-subscribed set P is chosen based on the low subscription count criteria (P_{sl}), we can see that all three algorithms perform equally by recommending lists from the auxiliary set. We also see that the member count plays an important role in deciding the list authority. If P is chosen such that the lists contain a large member count, then most recommended lists are from the users' direct subscription. On contrary, if P is chosen with low membership count all the algorithms tend to recommend the auxiliary lists. It is important to note that both in-degree and PageRank closely follows our model.

Finally, when we choose P based on high klout score (P_{kh}), the LIST-PAGERANK model recommends a majority of lists from users' direct subscription; thus, indicating that the individual authority of list members collectively contribute to the total list authority. Similar to the subscription characteristic, both the in-degree and PageRank tend to recommend the direct list subscription rather than the auxiliary list. The in-degree however seems to recommend more auxiliary lists when compared to the other two. In case of low klout score (P_{kl}), all three algorithms perform in similar fashion.

5 Related Work

Over the past few years, researchers have proposed various methods to overcome the problem of information overload in social networks. These studies can be classified into three main categories: (a) reorganization of user timeline in microblogs, (b) topic modelling, and (c) personalized recommendation.
Reorganization of user timeline: The research on time-

line reorganization aims to re-rank the timeline of users in microblogging network like Twitter. Feng et al. (Feng and Wang 2013) build a feature-aware factorization model that uses the graph containing nodes in the form of users, publishers and tweets. They build their model based on the notion that the tweet history reveals user's personal preference. Bernstein et al. (Bernstein et al. 2010) adopt a topic based technique for organizing twitter feeds. In their work, the tweets are transformed into queries for external search engine. The popular terms are then assigned as topics. Burgess et al. use Twitter lists to tackle the problem of timeline reorganization. Since lists implicitly denote the topical interests of *twitterers*, they propose a system called *Butterworth* that can automatically build twitter lists by leveraging user's social network and the content generated by friends. Our work is different from the ones mentioned above since it uses a novel *list based PageRank algorithm*. None of these works mention about the topic of list recommendation which forms a core part of our work.

Topic modeling The use of topic models in microblogging has been extensively studied by many authors. Ramage et al. (Ramage, Dumais, and Liebling 2010) presents a scalable implementation of labeled LDA. Phan et al. (Phan, Nguyen, and Horiguchi 2008) use the LDA topic model for building short and sparse text classifiers. (Abel et al. 2011) propose a URL recommendation system for Twitter users. According to the authors, the topics in Twitter are presented by different concepts that change over time. The concepts are built using a linguistic model that detects entities and mentions from users' tweets. Our topic modeling method uses the notion of dynamic temporal LDA which is not captured by the methods mentioned above.

User Recommendation Unlike timeline reorganization that restricts itself to the ranking of tweets, the user recommendation tackles the information overload problem by providing users with contents or users that are relevant to the user's interest. In (Chen et al. 2010), a URL recommendation system for twitter users is proposed which aims to recommend URLs by constructing a vector-of-words from users' tweets to measure their interest. In our previous work (Rakesh, Reddy, and Singh 2013), we developed a model that recommends geo-location based tweet summaries. Analyzing the tweet's content and social graph for recommending friends and followers have been studied by various researchers (Hannon, McCarthy, and Smyth 2011), (Armentano, Godoy, and Amandi 2012), (Hannon, Bennett, and Smyth 2010). There are very few studies that exploit the list feature in Twitter (Burgess et al. 2013; Yamaguchi, Amagasa, and Kitagawa 2011). In their studies, (Burgess et al. 2013) rank the tweets within the subscribed lists of users rather than recommending existing lists. In short, their work is similar to the ranking of user's timeline and hence it is quite different from our work. The models proposed in our work are similar to the ones described in (Weng et al. 2010) and (De Francisci Morales, Gionis, and Lucchese 2012). However, unlike these works, our paper leverages on the Twitter *lists* and temporal interest of users to design a new recommendation system.

6 Conclusions

As more and more users join social networking platforms like Twitter, facebook, foursquare etc., the data will be generated at an overwhelming pace, resulting in the problem of information overload. To overcome this problem, social networking sites have introduced the concept of *lists* that help users organize related information into a single bin. Despite being a powerful tool to organize related users and topics, it requires the laborious task of manually adding people who post about a similar topic. In this paper, we outlined two major problems. First, we showed that majority of users have sparse list subscriptions. Second, we showed that most lists have extremely low number of subscribers, which in turn means that they are inferior in their topical content. To overcome the first problem we introduced the `ListRec` model. We formulated this model as a linear combination of content, network and trendiness based weighting schemes, and estimated the parameters using a cyclic ridge regression algorithm. Our results showed that the `ListRec` model outperformed other base line models in all the performance measures. Furthermore, we showed the importance of using temporal topic model by leveraging our rich repository of temporally distributed streaming tweets. The results clearly showed that the user of dynamic temporal topic model (dDTM) over the conventional topic model (LDA) resulted in a superior recommendation.

To handle the second problem, we introduced a LIST-PAGERANK model that recommends auxiliary lists that are significantly better than the existing lists that are directly subscribed by the *twitterers*. To design this model, we introduced a new subscriber-member based relationship for the edges in the list network. Using empirical evaluation techniques, we showed that our model is efficient in recommending lists that contain members who are topically authoritative. We also showed that the recommended set of lists have high retweet and subscriber counts; thus, indicating it's topical dominance.

Acknowledgement

This work was supported in part by NSF grants IIS-1242304, IIS-1231742 and NIH grant R21CA175974.

References

Abel, F.; Gao, Q.; Houben, G.-J.; and Tao, K. 2011. Analyzing temporal dynamics in twitter profiles for personalized recommendations in the social web. In *Proceedings of the 3rd International Web Science Conference*, 2. ACM.

Armentano, M. G.; Godoy, D.; and Amandi, A. 2012. Topology-based recommendation of users in micro-blogging communities. *Journal of Computer Science and Technology* 27(3):624–634.

Bernstein, M. S.; Suh, B.; Hong, L.; Chen, J.; Kairam, S.; and Chi, E. H. 2010. Eddi: interactive topic-based browsing of social status streams. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, 303–312. ACM.

Blei, D. M., and Lafferty, J. D. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, 113–120. ACM.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3:993–1022.

Borgs, C.; Chayes, J.; Karrer, B.; Meeder, B.; Ravi, R.; Reagans, R.; and Sayedi, A. 2010. Game-theoretic models of information overload in social networks. In *Algorithms and Models for the Web-Graph*. Springer. 146–161.

Burgess, M.; Mazzia, A.; Adar, E.; and Cafarella, M. 2013. Leveraging noisy lists for social feed ranking. In *Seventh International AAAI Conference on Weblogs and Social Media*. ICWSM.

Chen, J.; Nairn, R.; Nelson, L.; Bernstein, M.; and Chi, E. 2010. Short and tweet: experiments on recommending content from information streams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1185–1194. ACM.

Chen, K.; Chen, T.; Zheng, G.; Jin, O.; Yao, E.; and Yu, Y. 2012. Collaborative personalized tweet recommendation. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, 661–670. ACM.

De Francisci Morales, G.; Gionis, A.; and Lucchese, C. 2012. From chatter to headlines: harnessing the real-time web for personalized news recommendation. In *Proceedings of the fifth ACM international conference on Web search and data mining*, 153–162. ACM.

Feng, W., and Wang, J. 2013. Retweet or not?: personalized tweet re-ranking. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 577–586. ACM.

Hannon, J.; Bennett, M.; and Smyth, B. 2010. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, 199–206. ACM.

Hannon, J.; McCarthy, K.; and Smyth, B. 2011. Finding useful users on twitter: twittomender the followee recommender. In *Advances in Information Retrieval*. Springer. 784–787.

Hong, L., and Davison, B. D. 2010. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, 80–88. ACM.

Hong, L.; Doumith, A. S.; and Davison, B. D. 2013. Co-factorization machines: modeling user interests and predicting individual decisions in twitter. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 557–566. ACM.

Järvelin, K., and Kekäläinen, J. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)* 20(4):422–446.

Kang, J. H., and Lerman, K. 2012. Using lists to measure homophily on twitter. In *AAAI workshop on Intelligent techniques for web personalization and recommendation*.

Kim, D.; Jo, Y.; and Moon, I.-C. 2010. Analysis of twitter lists as a potential source for discovering latent characteristics of users. In *CHI 2010 Workshop on Microblogging: What and How Can We Learn From It?* Citeseer.

Kwak, H.; Lee, C.; Park, H.; and Moon, S. 2010. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, 591–600. ACM.

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

Manjhi, A.; Shkapenyuk, V.; Dhamdhere, K.; and Olston, C. 2005. Finding (recently) frequent items in distributed data streams. In *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, 767–778. IEEE.

Mehrotra, R.; Sanner, S.; Buntine, W.; and Xie, L. 2013. Improving lda topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, 889–892. ACM.

Pennacchiotti, M.; Silvestri, F.; Vahabi, H.; and Venturini, R. 2012. Making your interests follow you on twitter. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, 165–174. ACM.

Phan, X.-H.; Nguyen, L.-M.; and Horiguchi, S. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, 91–100. ACM.

Qu, Z., and Liu, Y. 2011. Interactive group suggesting for twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, 519–523. Association for Computational Linguistics.

Rakesh, V.; Reddy, C. K.; and Singh, D. 2013. Location-specific tweet detection and topic summarization in twitter. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 1441–1444. ACM.

Ramage, D.; Dumais, S. T.; and Liebling, D. J. 2010. Characterizing microblogs with topic models. In *ICWSM*, 130–137.

Weng, J.; Lim, E.-P.; Jiang, J.; and He, Q. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, 261–270. ACM.

Yamaguchi, Y.; Amagasa, T.; and Kitagawa, H. 2011. Tag-based user topic discovery using twitter lists. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, 13–20. IEEE.