An Analysis of Topical Proximity in the Twitter Social Graph

Abstract. Standard approaches of information retrieval are increasingly complemented by social search even when it comes to rational information needs. Twitter, as a popular source of real-time information, plays an important role in this respect, as both the follower-followee graph and the many relationships among users provide a rich set of information pieces about the social network. However, many hidden factors must be considered if social data are to successfully support the search for high-quality information. Here we focus on one of these factors, namely the relationship between content similarity and social distance in the social network. We introduce a novel metric for measuring the topic similarity among twitter users and compare it to a standard text-based approach. Latent Dirichlet Allocation was applied to a one-per-user document collection to compute topic similarity. By comparing this metric at different hop distances in the social graph we investigated the utility of prominent features such as Retweets and Hashtags as predictors of similarity, and demonstrated the potential of topic proximity for friend recommendations.

Keywords: micro blogs, topical proximity, social network distance, friend recommendation

1 Introduction

The quality and relevance of information is crucial for a wide range of information-related activities, ranging from rational information retrieval to engagement in social interactions. Recently, recommendation systems have tried to improve the quality of search results by incorporating social signals, see for example [1] for an approach to developing a search engine on top of Twitter or the work of [2, 3] to incorporate social signals into mainstream search engines like Google. The objective with these approaches to social search is to increase the likelihood of addressing the searcher's genuine information needs by leveraging the content sharing and search experiences of their social networks in addition to more conventional index-based techniques. Social approaches have become especially relevant in the context of providing users with more *novel*, *diverse*, and *timely* results; see [4, 5] for recent trends in recommendation diversification. Also, feedback signals in social communities have been used to assess two types of signals per user independently, with respect to both of their roles as target and source of feedback respectively. This way, while processing feedback events in their temporal order, not only the quality of the user as content creator is evaluated, but also also the quality of the user as a content evaluator, see [6].

Conventional recommendation systems aim to proactively suggest relevant results to users based on the information consumption histories of similar, albeit anonymous, users. More recently, researchers have acknowledged that similarity alone does not guarantee relevant or optimal recommendations. In particular the importance of real-world user relationships has been highlighted as an important factor to leverage improved recommendation quality. For example, there has been considerable interest in modeling the reputation of users to bias future recommendations from users who are both relevant *and* reputable; see for example [7–9]. By leveraging the social web, research such as [10] explores the use of services like Twitter as a new source of item data and user opinions, showing how even this noisy signal can be used to make reliable recommendations.

In this paper, we quantify the extent to which social neighbors share interests in similar topics. We do this by a novel metric for topic similarity which is analyzed along star chain sequences emanating from a set of seed users. This is a first step towards the development of a concise framework for the evaluation of social recommendation against both rational and social criteria of quality. In this work we focus on Twitter. Twitter is an exceedingly popular microblogging platform and members typically choose their followee connections (friends) according to their topical interests rather than according to their social connections. Prior research in this area has investigated the prediction accuracy for followees based on latent topic similarity, see [11, 12]. We have a similar approach to the measurement of user similarity based on latent topics, but we are not interested in prediction accuracy. Rather, we aim to test whether and to which extent distance in a social network correlates with content similarity. In fact, the work of [11] can be viewed as a sufficient condition for the existence of differences in topic similarity between neighbors and non-neighbors. However, it would not directly support the notion of social recommendation, since the quality of the difference remains unclear. Maybe we chose our friends to be specifically diverse and to cover a wide range of topics. Clearly, social proximity facilitates content migration thereby increasing social similarity. But does this similarity extend to content, i.e. do people in close relationships talk about similar things? We will give some answers to this question as by-products of our research throughout the result section of this paper. However, our main concern and target is the proposition of a *topical proximity measure* and its validation by comparing its expressiveness to a baseline measure using term frequencies. As a consequence, we looked at a variety of methods to compute content similarity based on Latent Dirichlet Allocation, Term Frequency Vectors, and Pearson's Correlation and compared them with respect to network distance.

We base our investigation on a Twitter data set collected during November 2011 at the University of California. We took a reasonably sized sample of tweets per user as the source for a summary document, which was created in various ways; see Section 2.

The remainder of this paper is structured as follows: Section 2 describes our crawling algorithm and the various ways to build user summary documents from the resulting data. Next, we introduce our use of Latent Dirichlet Allocation (LDA) and resulting similarity measures in Section 3. Finally, after presenting detailed analysis of our results, we conclude with a discussion and suggest a number of opportunities for future work.

2 Data Crawl, Star Chaining, and User Models

Twitter is an interesting social platform because it combines the follower-followee social network structure with many dynamic sub-networks based on information flow. For example, mentions and retweets link users in ad-hoc networks as they discuss particular topics and events. We are interested in capturing topic associations from free-text content and from hashtags, and performing various similarity assessments based on properties in the underlying social structure.

Figure 1 shows an example of the structures crawled from the Twitter API, in this case for a single seed user. Starting from a set of these seed users, we created chains of length 3, i.e. we included all followees and followers of the seed users and both a followee and a fof (follower of the follower) for each followee. We only considered seed users for which at least 50 such chains could be found, with members across chains all distinct. In total, we crawled 5628 seed users with more than 20.000 followee 3-chains involving in total 22.549 users. We also collected data about mentions, re-tweets and other statistical information about users such as profile age and number of friends and followers.

Obviously, we were not able to crawl all friends, friends of friends, and friends of friends of friends in our data set. Rather, we started with a seed crawl using the Twitter API and searched for particular keywords, here *Libya*, to allow for a good network density among the Tweeters. After this initial crawl, we had 33740 seed users and considered them as a network sample. We then crawled the other tweets for those users and all of their followees. Using this data, we obtained our 3-chains by a heuristic star chain algorithm; details below. Note, we can guarantee that all of the 1-hop and 2-hop members of the chains we created are indeed shortest distance to the seed user, but we cannot give such a guarantee for the 3-hop chain members.

In order to give such a guarantee, we have to get all friends of friends for all seed users, which will require millions of calls to the Twitter API resulting in billions of results. The details of the algorithm used to obtain the chains is given in Figure 2.

For each user, we subsequently generated a *user profile document* based on 50 randomly selected tweets from the crawl. Depending on the exact method, we either included retweets, excluded retweets, or selected only retweets. Each user is represented as the collection of words in their tweets, after filtering mechanisms are applied. Filtering is a crucial part of our research since we want to investigate the influence and use of certain elements of language and communication within tweets. Therefore, we give special consideration to *hashtags* as well, in addition to *retweets*. Hashtags are named entities that are used for content tagging and search both within twitter and by external services such as Listorious. Therefore we can either consider them as part of the text (by simply removing the hashtag symbol #), or leave them out, or create the summary document per tweet user solely from them. All in all, we considered five different methods as depicted in Table 1.

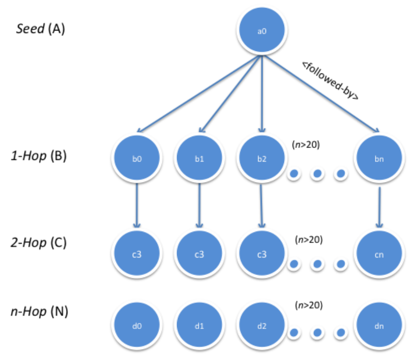


Fig. 1. Example of a "star-chain" pattern from the data crawl. Topic similarity is computed between each hop A-D. This view is for a single user. Note, we do not illustrate the 3-Hop (D) here, even though we extended our analysis to include the 3-Hop.

3 Similarity Measures

Based on the user profile documents described earlier, we computed pairwise similarity among all pairs of users and their followees. We used simple Pearson correlation over the term frequency vectors as a *text-based* baseline and compared it to our *topic-similarity* measure based on Latent Dirichlet Allocation (LDA), see Section 4. We used an implementation of Blei's LDA algorithm [13] from the Stanford Topic Detection Toolbox[1], which provides multiple different implementations of Latent Dirichlet Allocation, see [13, 14]. We removed terms consisting of less than 3 characters, occurring in less than 3 profiles and the 30 most common terms. Stop words were also removed. We used 20 topics. Once the probability vectors across topics are computed for each user, we used Pearson's correlation between any pair of users to compute the pairwise similarity with

[1] http://nlp.stanford.edu/software/tmt/tmt-0.4/

```
procedure getStarChains(Users, Links)
    chains = ∅
    for all seed ∈ Users do
        seedChains = ∅
        F1 = Links(seed)
        2A = Links(F1) − F1
        2U = ∅
        3U = ∅
        for all f1 ∈ F1 do
            F2 = Links(f1) − {seed} − F1 − 2U
            for all f2 ∈ F2 do
                found = false
                F3 = Links(f2) − {seed} − F1 − 2A − 3U
                for all f3 ∈ F3 do
                    found = true
                    seedChains ← (seed, f1, f2, f3)
                    2U ← f2
                    3U ← f3
                    break
                end for
                if found then
                    break
                end if
            end for
        end for
        if Count(seedChains) >= 50 then
            chains = chains ∪ seedChains
        end if
    end for
    return chains
end procedure
```

Fig. 2. The Star Chaining Algorithm: The outermost loop iterates through all the seed users. The three inner loops are meant to find a sequence of neighbors without repetition and without shortcuts. Fx is the set of candidate users for the x-th position of the chain for $x \in \{1, 2, 3\}$, $2A$ is the set of all friends of a friend for one particular seed user, and xU is the set of users previously used at the x-th position of the chain for $x \in \{2, 3\}$. Note also, that the algorithm does simple backtracking, i.e. if it does not find a third neighbor for a particular second neighbor, then it will try other second neighbors until a third neighbor is found, if possible. However, since the selection of chains is arbitrary, the algorithm does not guarantee the maximum number of chains per seed user to be found.

Table 1. Representation of Twitter Users

No	User Representation	ID	Description
1	All Text	All	The collection of all words in the selection of tweets, but with the hashtag symbol # removed
2	No Retweets	All-RT	The collection of all words in the selection of tweets without retweets, but with the hashtag symbol # removed
3	Only Retweets	RT	The collection of all words in the selection of tweets without retweets, but with the hashtag symbol # removed
4	No Hashtags	All-HT	The collection of all words in *All*, but without hashtags
5	No Hashtags, No Retweets	Filt	The collection of all words in *All-RT*, but without hashtags, fully filtered
6	Simple Hashtags	HT	Only the set of hashtags in *All*
7	Simple Hashtags, No Retweets	HT-RT	Only the set of hashtags in *All-RT*

respect to topics. In general the term lists (topics) output by the LDA algorithm were representative of human-understandable themes. Table 2 shows example topics from our data set. The two methods are summarized in Table 3.

Table 2. Example LDA term-lists (topics) mined from the Twitter API.

Comment	*Topic*
Libya Crisis	gaddafi tripoli nato libyan feb17
Social	today photo stories facebook google
US Elections	gop cain president romney perry
UK Related	bbc london guardian cameron telegraph

4 LDA-based Topic Similarity

We based our novel topic similarity measure on Latent Dirichlet Allocation (LDA) [13], a simple probabilistic topic model. With LDA, a topic is modeled as a distribution over a fixed vocabulary, while each word in a document belongs to a particular topic. Moreover, each document is associated with each topic in a certain way, and the model assumes that documents are focussing on particular topics. The LDA model is then trained with a set of documents to assign the probability distribution over the fixed vocabulary in such a way that the resulting topic-document associations fulfill the model and the likelihood of the documents to occur under this model is maximized.

Here we consider users as being represented by documents, so the result of training an LDA model for our set of user profile documents is a vector

Table 3. Methods to Measure Content Similarity

Method	ID	Description
Topical Proximity	LDA	Compute User-Topic Associations for a fixed Set of Topics based on Latent Dirichlet Allocation (LDA) on the user profile documents, then compute similarity as Pearson's Correlation
Text Similarity	TF	Compute term frequency vectors for the user profile documents, then compute similarity as Pearson's Correlation

containing topic associations across all topics for each user. Note, the number of topics must be fixed prior to training for standard LDA. It is a matter of future research whether standard LDA techniques are really a good match for per-User Microblog data.

Now we apply Pearson's correlation among these topic vectors to obtain User-to-User topic similarity in our data corpus.

5 Friend Recommendation - A Potential Application

Whether or not topical proximity can be used for friend recommendation, that depends on many factors. To get a first understanding about the potential of this application, we conducted the following experiment. For each seed user, we choose 20 neighbors from the hop-1 and the hop-2 layers (10 each). We then recommended the top-10 similar users among the whole list of 20 to the seed user as hop-1 neighbors and measured *B*-*C* $\rightarrow$ *B precision*, i.e. the ratio of recommended neighbors that are indeed in the hop-1 layer. Obviously, this experiment is somewhat flawed by the actual social influence among neighbors, but it may give some qualitative insight in the potential of our various methods to act as a good recommender.

6 Results

To analyze the influence of network distance on topic simlarity between Twitter users, we begin by averaging the topic similarities according to hop distances. In Figure 1, these are $sim(A, B)$, $sim(A, C)$, $sim(A, D)$, and $sim(A, N)$ respectively, where N is an arbitrary set of Twitter profiles randomly taken from our crawled set of seed users. In addition to the average similarity values, we also computed *B*-*C* $\rightarrow$ *B precision*, as discussed in Section 5, to provide a more complete picture. Results are shown for all methods in Table 4 and discussed in the sequel.

6.1 Similarity Analysis of Hop Distance

Table 4 shows a table of 9 experimental conditions. The second column represents the method for user representation from Table 1 above, while column three

Table 4. Results of Topic-based Similarity Analysis

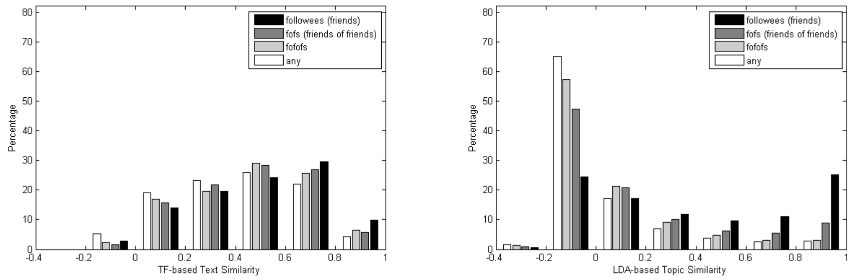
ID	*Represen-tation*	*Method*	*sim (A,B)*	*sim (A,C)*	*sim (A,D)*	*sim (A,N)*	*B-C → B Precision*
A	All	LDA	0.3894	0.1550	0.0618	0.0326	0.7225
B	All-RT	LDA	0.3268	0.1500	0.0769	0.0454	0.6912
C	HT	LDA	0.2490	0.0850	0.0203	0.0089	0.5965
D	HT-RT	LDA	0.1847	0.0644	0.0123	0.0067	0.6399
E	All-HT	LDA	0.3841	0.1545	0.0602	0.0320	0.7306
F	Filt	LDA	0.3129	0.1459	0.0789	0.0446	0.7077
G	All	TF	0.4587	0.4323	0.4218	0.3834	0.5147
H	RT	TF	0.4964	0.5373	0.5000	0.3990	0.5258
I	HT	TF	0.0712	-0.0023	-0.0745	-0.0607	0.6091

represents the method for computing content similarity, i.e. either *text-based* (TF - Term Frequency) or *topic-based* (LDA - Latent Dirichlet Allocation). The four rightmost columns represent the average similarity scores between the seed user and the star-chain. These scores are calculated by a simple application of Pearson's correlation over the topic associations between the seed profile (a_0) in Figure 1, and each profile in the one-hop layer ($B = b_0 ... b_n$ in Figure 1), the two hop layer ($C = c_0 ... c_n$ in Figure 1) and finally the n-hop layer (the disconnected component of Figure 1). Averages are calculated across all chains irrespective of the seed user they belong to.

Results vary significantly depending on the use of Retweets, the method to create the user profile document, and the chosen similarity measure, albeit the removal of Hashtags does not seem to be important, see cases E and F which are not very different from cases A and B. For the same reason, we skipped computing their corresponding *text-based* (TF) similarity measures. In the sequel, we will discuss the results in a deeper fashion and extend the analysis in some cases. Even though our main interest is the decay for topical similarity across hop distances, we will first take a look at the differences between baseline text-based similarity and topic similarity.

6.2 Topic vs. Text Similarity

We look at the unfiltered cases all, LDA and all, TF to examine the differences between LDA-based topic similarity and TF-based text-similarity. The topic similarity case exhibits a score of 0.3894 for the hop-1 correlation $simA, B$ decaying to 0.1550, and 0.0618 along the chain ($simA, C$ and $simA, D$) and to 0.0318 with respect to a random node among the seed users. For the text similarity case, we have quite different numbers. However, the decay along the chain is obvious in both cases. For all, TF, the values are falling for each pair along the chain, confirmed by a one-sided t-test with significant level 0.05. A deep analysis of the differences is given in Figure 3. Here we looked at the distribution of similarity values, basically a normalized histogram of the similarity values with 7 bins of size 0.2 between -0.4 and 1. Similarity values lower than -0.4 occurred rarely and have been omitted here for presentational clarity.



(a) All, TF (b) All, LDA

Fig. 3. Text Similarity (TF-based) and Topic Similarity (LDA-based) for Conditions A and G

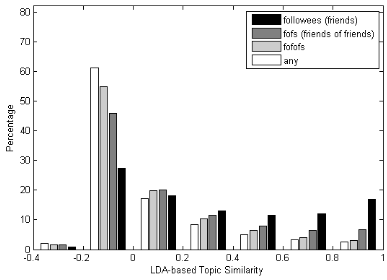
The difference in signal clarity between *text similarity* and topic similarity is obvious.

6.3 Retweets and Hashtags

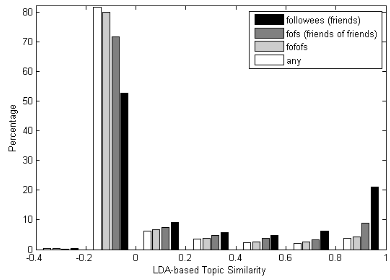
Looking at the same conditions but this time with no retweets ($All - RT, LDA$), we see a slight decrease of -16% in the hop-1 similarity score $simA, B$ which falls to 0.3268 but there is no large difference at hop-2 and hop-3, so obviously Retweets have no influence on topic similarity beyond direct neighbors, as expected. Again, there is no difference when we apply the hashtag filter, see case F. The use of hashtags alone produced lower correlations, see cases C and D in Table 4. This could be credited to the smaller amount of available text data as Blei reports in [13] that LDA requires large amounts of text as input to function well. However it is more likely that hastags are free of shared vocabulary or irrelevant chat topics and therefore provide a better means to see the true topical relationships between neighbors, especially if we remove retweets as in the $false, HT, LDA$ case.

In order to understand the exact difference between the methods according to different hop distances, we produced distribution graphs for conditions A-D, see Figure 4.

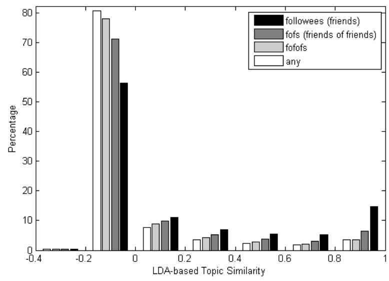
As expected, 1-hop neighbors are leaning towards higher similarities. There is a fraction of 1-hop neighbors with a very high correlation and this may indicate selectiveness in choosing your friends according to particular topics, but it could also have other reasons. Neighbors 3-hops away from the seed user did not exhibit a big similarity difference from the randomly chosen (n-hop) set.



(a) All-RT, LDA

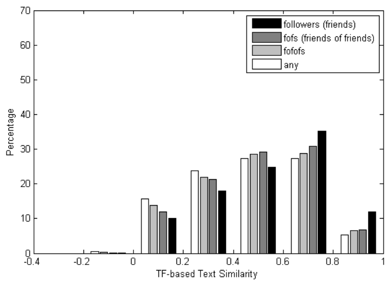


(b) HT, LDA

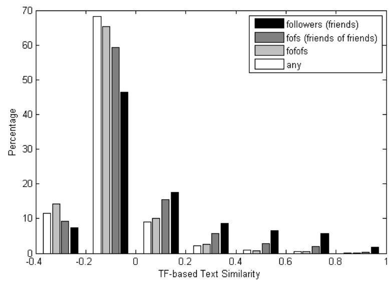


(c) HT-RT, LDA

Fig. 4. Network-based Comparison of Topic Similarity (LDA-based) for Conditions B, C, and D



(a) RT, TF



(b) HT, TF

Fig. 5. Network-based Comparison of Text Similarity (TF-based) for Conditions H and I

6.4 Analysis of Decay Patterns

To analyze whether or not the decay pattern is different for different groups of users, we classified groups according to their average *sim (A,B)* value. Without loss of generality, we chose the $All - RT, LDA$ condition for this analysis. We classified the users into three groups depending on the mean value of their Hop-1 topic similarity value. The results are shown in Table 5. The left two columns define the min and max of the range of mean *sim (A,B)* values for the group, the third column is the count of users in our result data set, and the third and fourth column are the averages *sim (A,B)* and *sim (A,C)* across that group of users. Note, the aggregated results in the bottom row are not exactly the same with Table 4, due to a different aggregation technique and due to different processing, i.e. we needed to remove an entire chain here whenever one similarity value was missing. The last column is the *decay* factor, i.e. the percentage of the value in column 5 with respect to the value in column 4.

Table 5. Decay Pattern for different User Groups

Group	Min	Max	Count	*sim (A,B)*	*sim (A,C)*	*decay*
low	-1	0.1	2192	0.0559	0.0758	1.3556
medium	0.1	0.4	2271	0.2976	0.1426	0.4790
high	0.4	1	2963	0.6117	0.2200	0.3597
all	-1	1	7426	0.3369	0.1507	0.4473

Interestingly, the *low* group seems to have a random distribution of topic similarity across its neighbors, i.e. their similarity is not very different from the similarity of Hop-2 neighbors. Users with high Hop-1 topic similarity also have a higher Hop-2 topic similarity.

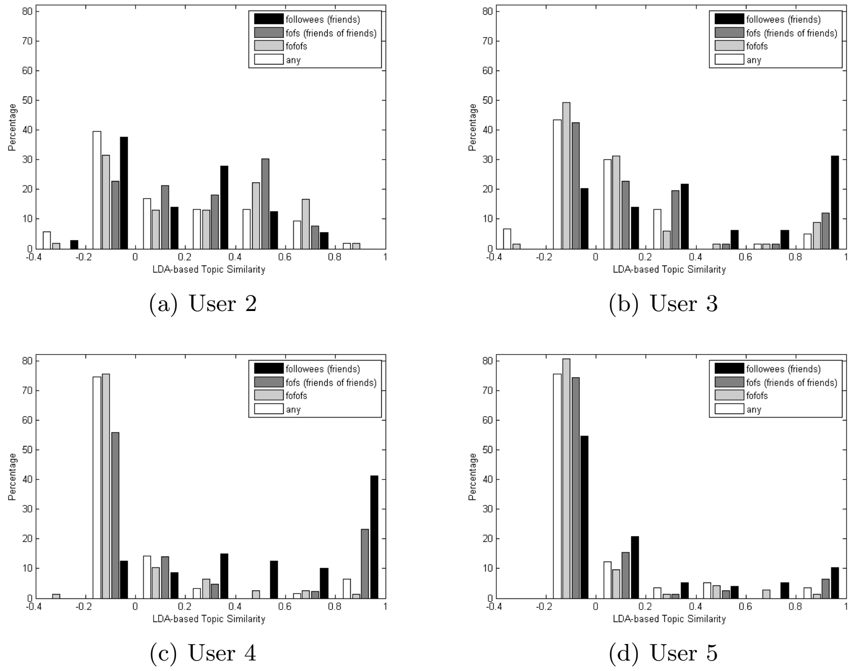
6.5 Single User Analysis

Our star-chaining approach facilitates a further, more fine grained data analysis of individual seed users. We randomly select 5 such users from the $All-RT, LDA$ case and show their results in Table 6. The new second column is the range of the chain count that was used for computing the *Hop-1*, *Hop-2*, *Hop-3*, and *Any* averages in columns $3-6$. Again we illustrate the distributions across similarity values as before, omitting the less interesting case of *User 1*, see Figure 6.

Table 6. Topic-based Similarity Analysis for Individual Users.

User ID	#	*sim (A,B)*	*sim (A,C)*	*sim (A,D)*	*sim (A,N)*
1	26-38	0.2522	0.0445	0.0040	-0.0665
2	54-72	0.1570	0.2499	0.2455	0.1491
3	64-67	0.4028	0.1600	0.0980	0.0506
4	78-86	0.5657	0.2086	-0.0076	0.0290
5	72-77	0.1429	0.0224	-0.0155	0.0162

The following observations can be made, some of which would not be possible with text-based similarity analysis only:



(a) User 2 (b) User 3

(c) User 4 (d) User 5

Fig. 6. Comparison of Individual Users

- *User 2* does not seem to be topically selective with respect to his followees, since there is no big difference between Hop-1 and the other distances outside the control of the user.
- *User 3* is very similar to some of his followees, but this might be a result of direct Retweets, as there is no clear difference between *Hop-2*, *Hop-3*, and *Any*.
- *User 4* is similar to *User 3*, this user is very similar to some of his followees, but this effect does not decay very quickly for *Hop-2*. Probably this is a user with a special interest, an assumption which would be supported by the fact that *Hop-2* and *Any* neighbors are very likely to share a low topic similarity.
- *User 5* is similar to *User 4*, this user probably has a special interest, because of the high likelihood to share a low topic similarity. However, different from *User 4* and similar to *User 2*, this user's *Hop-1* similarity distribution is similar to the other three hop distances.

6.6 Potential as a Friend Recommender

B-C $\rightarrow$ *B precision* as shown in Table 6 is just an indicator for the decay of topical proximity between hop-1 and hop-2 neighbors. Nonetheless, it is remarkable that the values are much higher for LDA-based topical proximity than for term-frequency-based text similarity. For text similarity methods, hashtags are better than the full text, but for LDA-based proximity the opposite is true, probably because hashtags are easily adopted among neighbors and therefore provide proximity clues that are automatically detected by LDA.

6.7 Discussion of Results

The results clearly indicate the power and applicability of LDA-based Pearson's correlation as a measure for topical proximity in the Twitter Social Graph. Especially for large collections of data, topic similarity signals are much stronger than text similarity signals and provide a deeper resolution for a detailed analysis. We showed that 1-Hop similarity is correlated with 2-Hop similarity, thereby supporting research who is trying to recommend friends from the nearby neighborhood. Moreover, we showed that topic similarity with full texts is a stronger predictor of 1-Hop neighbors than hashtags, no matter whether we use LDA-based Pearson's correlation or term-based similarity measures for the latter. However users are apparently not all made equal. In Section 6.5, we demonstrate by detailed analysis of 5 random users, that there is a huge variety of different case which could easily be distinguished by the presented research, and which is one of venues to be followed up in future research. We showed that the removal of hashtags from the tweets prior to processing only has a very minor influence on topical proximity.

7 Conclusion and Future Work

In this paper we have presented a deep analysis of topic and content similarity along the twitter social graph. By computing similarity between users based on their underlying topical proximity, and comparing it against a text-based term frequency approach, we quantify the notion of topical similarity among micro bloggers and demonstrate distinct similarity patterns through a novel use of LDA.

There are a number of implications of the results of this study for Twitter users and application designers, particularly with respect to applications like social recommender systems [15] wherein there is usually an accompanying set of social links to augment traditional nearest-neighbor selection strategies. We believe that our approach and findings can be applied across any corpus of text-based content that has an associated underlying network structure. The results of our topic-similarity analysis show that 1) there is generally a strong LDA topic similarity between direct neighbors in the social graph, and 2) this value decays to a standard value usually by the third hop, and in some cases at only two hops in the Twitter graph. We showed that and how those findings can easily be used to develop an LDA-based topical friend recommender.

Future research is directed towards the evolution of a concise framework for automatic measurement of various qualities relevant to the user in a social network. The exploitation of this framework for building applications and evaluation of recommender systems is an obvious opportunity for us and others alike.

References

1. Phelan, O., McCarthy, K., Bennett, M., Smyth, B.: Terms of a feather: Content-based news recommendation and discovery using twitter. In Clough, P., Foley, C., Gurrin, C., Jones, G.J.F., Kraaij, W., Lee, H., Murdock, V., eds.: ECIR. Volume 6611 of Lecture Notes in Computer Science., Springer (2011) 448–459
2. Smyth, B., Briggs, P., Coyle, M., O'Mahony, M.P.: Google shared. a case-study in social search. In: User Modeling, Adaptation and Personalization. (2009) 283–294
3. McNally, K., O'Mahony, M.P., Smyth, B., Coyle, M., Briggs, P.: Social and collaborative web search: an evaluation study. In: Proceedings of the 16th international conference on Intelligent user interfaces. IUI '11, New York, NY, USA, ACM (2011) 387–390
4. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: Proceedings of the Second ACM International Conference on Web Search and Data Mining. WSDM '09, New York, NY, USA, ACM (2009) 5–14
5. Vargas, S., Castells, P.: Rank and relevance in novelty and diversity metrics for recommender systems. In: Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, October 23-27, 2011. (2011) 109–116
6. Schaal, M., Fidan, G., Müller, R.M., Dagli, O.: Quality Assessment in the Blog Space. The Learning Organization **17**(6) (2010) 529–536
7. Bourke, S., McCarthy, K., Smyth, B.: Power to the people: exploring neighbourhood formations in social recommender system. In: Proceedings of the fifth ACM conference on Recommender systems. RecSys '11, New York, NY, USA, ACM (2011) 337–340
8. Jamali, M., Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks. In: Proceedings of the fourth ACM conference on Recommender systems - RecSys '10, New York, New York, USA, ACM Press (September 2010) 135
9. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: Proceedings of the 10th international conference on Intelligent user interfaces. IUI '05, New York, NY, USA, ACM (2005) 167–174
10. Esparza, Sandra Garcia, O'Mahony, Michael P., Smyth, Barry: Effective Product Recommendation using the Real-Time Web. In Bramer, M., Petridis, M., Hopgood, A., eds.: Research and Development in Intelligent Systems XXVII, London, Springer (2011) Proceedings of AI-2010.
11. Puniyani, K., Eisenstein, J., Cohen, S., Xing, E.P.: Social links from latent topics in microblogs. In: Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media. WSA '10, Stroudsburg, PA, USA, Association for Computational Linguistics (2010) 19–20
12. Kang, B., O'Donovan, J., Höllerer, T.: Modeling topic specific credibility on twitter. In: IUI. (2012) 179–188
13. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3** (March 2003) 993–1022
14. Gretarsson, B., O'Donovan, J., Bostandjiev, S., Höllerer, T., Asuncion, A.U., Newman, D., Smyth, P.: Topicnets: Visual analysis of large text corpora with topic modeling. ACM TIST **3**(2) (2012) 23
15. Woerndl, W., Groh, G.: Utilizing physical and social context to improve recommender systems. In: Web Intelligence/IAT Workshops. (2007) 123–128