Using Tag Recommendations to Homogenize Folksonomies in Microblogging Environments

Eva Zangerle, Wolfgang Gassler, Günther Specht

Databases and Information Systems
Institute of Computer Science
University of Innsbruck, Austria
firstname.lastname@uibk.ac.at

Abstract. Microblogging applications such as Twitter are experiencing tremendous success. Twitter users use hashtags to categorize posted messages which aim at bringing order to the chaos of the Twittersphere. However, the percentage of messages including hashtags is very small and the used hashtags are very heterogeneous as hashtags may be chosen freely and may consist of any arbitrary combination of characters. This heterogeneity and the lack of use of hashtags lead to significant drawbacks in regards of the search functionality as messages are not categorized in a homogeneous way. In this paper we present an approach for the recommendation of hashtags suitable for the tweet the user currently enters which aims at creating a more homogeneous set of hashtags. Furthermore, users are encouraged to using hashtags as they are provided with suitable recommendations for hashtags.

1 Introduction

Microblogging has become immensely popular throughout the last years. Twitter, the most successful platform for microblogging, is experiencing tremendous popularity on the web. Essentially, microblogging allows users to post messages on the Twitter platform which are at most 140 characters long. These posted messages – also known as tweets – are available to the public. Users are able to "follow" other users, which basically means that if user A follows user B (the followee), user A subscribes to the feed of tweets of user B. These messages are then added to the user's timeline (overview about his own tweets and the tweets of his followees) which enables him to always be up-to-date with the followee's tweets. Considering the fact that currently about 140,000,000 Twitter messages are posted every day, it becomes clear that the data posted is very diverse and heterogeneous. Therefore, Twitter users themselves started to manually categorize and classify their tweets – they started to use so-called *hashtags* as a part of the message. The only requirement for a hashtag is that it has to be preceded by a hash symbol #, like e.g. in the hashtags `#apple`, `#elections` or `#obama`. There are no further restrictions in regards of the syntax or semantics of hashtags, which makes them a very convenient, easy-to-use way of categorizing

tweets. Most importantly, hashtags can be used for searching messages, following a certain thread or topic and therefore mark a set of tweets focusing on a certain topic described by the hashtag. Hence, the use of appropriate hashtags is crucial for the popularity of a message in regards of how quickly messages concerning a certain topic can be found. Therefore, hashtags can also be seen as a way to give a certain amount of "context" to a tweet. However, choosing the best hashtags for a certain message can be a difficult task. Hence, users often feel forced to use multiple hashtags having the same meaning (synonyms), like e.g. for tweets regarding the SocInfo conference, one could use `#socinfo`, `#socinfo2011` and `#socinfo11`. The usage of multiple synonymous hashtags decreases the possible length of the actual content of the tweet as only 140 characters including hashtags are allowed per tweet. Furthermore, the usage of synonyms also motivates other users to cram their messages with hashtags to cover as many searches as possible. To avoid such a proliferation of hashtags, for example hashtags concerning a certain event are often predefined and propagated to all its participants in order to ensure that the hashtags used for tweets regarding this event are homogeneous. This often leads event organizers (e.g. of conferences) to announce an "official" hashtag. E.g. Tim O'Reilly (@timoreilly) posted on 2011-03-05: `At Wired Disruptive by Design conference, no hashtag announced. Hmmm.`. Such scenarios could easily be avoided if the tag vocabulary of the folksonomy is kept homogeneous which basically implies that no synonymous hashtags are used.
In this paper we present an approach aiming at supporting the user and creating a more homogeneous set of hashtags within the Twittersphere by facilitating a recommender system for the suggestion of suitable hashtags to the users. We show how the computation of hashtags can be facilitated and prove that this approach is able to provide the user with suitable hashtag recommendations.
The remainder of this paper is organized as follows. Section 2 outlines the characteristics of the data set underlying our evaluations. Section 3 is concerned with the algorithms underlying our approach. Section 4 features the evaluation of our approach and Section 5 describes work closely related to our approach. The paper concludes with final remarks in Section 6.

2 Used Dataset for Recommendations

The approach presented in this paper and its evaluation are based on an underlying data set of tweets which is used to compute the hastag recommendations. As there are no large Twitter datasets publicly available , we had to crawl tweets in order to build up such a database. The crawling of Twitter data has been constrained significantly by the abolishment of so-called Whitelisting. Whitelisting allowed users to query the Twitter API without any restrictions. Currently, the Twitter API only allows 350 requests per hour, each call returning about 100 tweets on average. The dataset was crawled by using the search API. As input for these search calls, we made use of an English dictionary consisting of more than 32,000 words. We used each of these words as input for the search process and stored the search results. This strategy enabled us to crawl about 18 million tweets between July 2010 and April 2011. Only 20% of these messages contained hashtags. Further details about the characteristics of the data set can be found in Table 1.

Characteristic	**Value**	**Percentage**
Crawled messages total	18,731,880	100%
Messages containing at least one hashtag	3,753,927	20%
Messages containing no hashtags	14,977,953	80%
Retweets	2,970,964	16%
Direct messages	3,565,455	19%
Hashtags usages total	5,968,571	–
Hashtags distinct	585,140	–
Average number of hashtags per message	1.5932	–
Maximum number of hashtags per message	23	–
Hashtags occurring < 5 times in total	502,172	–
Hashtags occurring < 3 times in total	452,687	–
Hashtags occurring only once	377,691	–

Table 1. Overview about the Tweet Data Set

3 Hashtag Recommendations

The recommendation of hashtags supports the user during the process of creating a new tweet. While the user is typing, hashtags appropriate for the already entered message are computed on the fly. With every new keystroke, the recommendations are recomputed and get refined. Due to the fact that both the cognition of the user and the space available for displaying the recommendations is limited, the shown size of the set of suggested hashtags is restricted. In most cases a set of 5-10 recommendations is most appropriate which also corresponds to the capacity of short-term memory (Miller, 1956). Therefore the top-k recommendations are shown to the user, where k denotes the size of the set of recommended hashtags. The value k was chosen between 1 and 10 in our evaluation. For a given tweet (or part of it), the computation of these recommendation for suitable hashtags based on the underlying data set comprises the following steps which are also illustrated in Figure 1.

1. For a given input tweet (or a part of it), retrieve the most similar messages featuring hashtags from the data set.

2. Extract the hashtags contained in the top-k similar messages. These hashtags constitute the hashtag recommendation candidate set.
3. Rank the recommendation candidates, computed in step 2 according to the ranking methods proposed in this paper.
4. Present the top-k ranked hashtags to the user.

These steps are described in detail in the following sections.

Fig. 1. Workflow: Hashtag Recommendation Computation

3.1 Similarity of Messages

Retrieving the set of k most similar messages to the input (query) tweet is the first step in computing recommendations. The similarity between the input tweet and the messages within the data set is computed by the cosine similarity of the tf/idf weighted term vectors. The messages within the data set are ranked according to this similarity measure and the top-k messages ($k = 500$ in our evaluations) are used for the further computation of recommendations as these most similar messages are most likely to contain suitable hashtags for the current input message. Therefore, the hashtags contained in these messages are extracted. These hashtags are referred to as hashtag recommendation candidates throughout the remainder of this paper.

3.2 Ranking

The ranking of the hashtag recommendation candidates is a crucial part of the recommendation process as only the top-k (with k between 5 and 10) hashtags are shown to the user. Therefore, we propose four basic ranking methods for the recommendation of hashtags. These ranking methods are either based on the hashtags themselves (TimeRank, RecCountRank, PopularityRank) or the messages where the tweets are embedded in (SimilarityRank).

– SimRank (1) - this ranking method is based on the similarity values of the input tweet t_{input} and the tweets containing the hashtag recommendation candidates $\mathcal{C}_T$. The cosine similarity has to be computed for every term within the input tweet and are used for the ranking of the recommendation candidates.
– TimeRank (2) - this ranking method is considering the recency of the usage of the hashtag recommendation candidates. The more recent a certain hashtag has been used, the higher its ranking. This ranking enables the detection and prioritization of currently trending hashtags (most probably about trending topics) which have been used only recently.

– RecCountRank (3) - the recommended-count-rank is based on the popularity of hashtags within the hashtag recommendation candidate set. This basically means that the more similar messages contain a certain hashtag, the more suitable the hashtag might be.
– PopRank (4) - the popularity-rank is based on the global popularity of hashtags within the whole underlying data set. As only a few hashtags are used at a high frequency, it is likely that such a popular hashtag matches the tweet entered by the user. Therefore, ranking the overall most popular hashtags from within the candidate set higher is also a suitable approach for the ranking of hashtags.

The ranking methods are formally described in the following equations, where $\mathcal{T}$ is the crawled data set containing all tweets and $\mathcal{C}_T$ is the candidate consisting of all top-k tweets regarding the similarity measure to the input string. $\mathcal{C}_H$ denotes the set of all extracted hashtags from the set $\mathcal{C}_T$. The function $contains(t, h)$ returns 1 if the specified hashtag h is present in the specified message t and 0 if it cannot be found in the message text. The function $now()$ returns the current UNIX-timestamp and $createdAt(t)$ corresponds to the timestamp the respective tweet t was created.

$$sim(t_{input}, t_c) = \frac{V(t_{input}) \cdot V(t_c)}{\parallel V(t_{input}) \parallel \parallel V(t_c) \parallel} \qquad foreach\ t_c \in \mathcal{C}_T, \tag{1}$$

where $V(t_{input})$ and $V(t_c)$ are the weighted term vectors of t_{input} resp. t_c

$$timeDiff(t_c) = now() - createdAt(t_c) \qquad \text{for each } t_c \in \mathcal{C}_T \tag{2}$$

$$recCount(h) = \sum_c contains(t_c, h) \qquad \text{with } \ t_c \in \mathcal{C}_T \tag{3}$$

$$pop(h) = \sum_i contains(t_i, h) \qquad \text{with } \ t_i \in \mathcal{T} \tag{4}$$

After the computation of the sim, $timeDiff$, $recCount$ and pop values, all suitable hashtag candidates of set $\mathcal{C}_H$ are subsequently ranked in descending order to compute the final ranking.

Beside these basic ranking algorithms, we propose to use hybrid ranking methods which are based on the presented basic ranking algorithms. The combination of two ranking methods is computed by by the following formula:

$$hybrid(r1, r2) = \alpha * r1 + (1 - \alpha) * r2 \tag{5}$$

where α is the weight coefficient determining the weight of the respective ranking within the hybrid rank. $r1$ and $r2$ are normalized to be in the range of $[0, 1]$ and can therefore be combined to a hybrid rank.

4 Evaluation

The evaluations were conducted based on a prototype of the approach which was implemented in Java on top of a Lucene fulltext index. As a data set based on which the evaluations were performed on, we used the data set described in Section 2. This implies that our Lucene index kept 3.75 million tweets. The evaluation was performed on a Quad-Core machine with 8 GB of RAM on CentOS release 5.1.

Essentially, we performed leave-one-out tests on the collected tweets in order to evaluate our approach. For this purpose, we arbitrarily chose 10.000 sample tweets from the data set. For our tests we only use tweets which contain less than 6 hashtags to exlude possible spam messages. Furthermore, we did not use any retweets or messages which are present several times in the dataset for the evaluation as these would lead to hashtag recommendations based on identical messages and would therefore distort our evaluation. Such a leave-one-out test consists of the following steps which were performed for each of the 10.000 test-tweets:

1. Remove the hashtags occurring in the test-tweet.
2. Remove the test-tweet from the index (underlying dataset) as leaving the original tweet in the index would lead to a perfect match when searching for similar messages. Therefore, also the original hashtags would be recommended based on the same tweet.
3. Use the test-tweet (without hashtags) or a part of the message as the input string for the recommendation computation algorithm.
4. Compute the hashtag recommendations using the recommendation approach including the different ranking methods introduced in section 3.
5. Evaluate the resulting hashtag recommendations in comparison to the originally used hashtags based on the measures described Section 4.1.

In order to determine the quality and suitability of the recommendations of hashtags provided to the users, we chose to apply the traditional IR-metrics recall, precision and F-measure (also known as F1-score). As a hashtag recommendation system should be aiming at providing the user with an optimal number of correct tags, the recall value is the most important quality measure for our approach.

4.1 Recall and Precision, F-Measure

Figure 2 shows the top-k ($k = 1, 2, ..., 10$) plot of the recall values of the four basic ranking methods. The good performance of the SimilarityRank can be explained by the fact that the message in which the hashtag recommendation candidate is embedded in is directly related to the relevancy of the hashtag. The other ranking methods are based on time or (global) hashtag popularity which are only loosely coupled to the hashtag and the message it is contained in. It can be seen that already five shown hashtags are sufficient to get a reasonable recall value of about 35% and therefore allow to build a lightweight recommendation interface without overwhelming the user by too many recommendations. The increment of the number of shown hashtags k showed very slight improvements regarding the recall value.

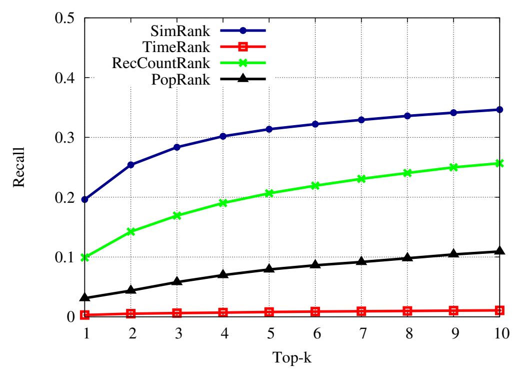

Fig. 2. Top-k Recall for k=[1..10] for the Basic Ranking Methods

As for the hybrid ranking approaches, we chose to evaluate these in regards of their recall, precision and F-measure. The SimilarityRank method proved to be the ranking method performing best throughout our evaluations. Therefore, we chose to combine the other ranking methods proposed in this paper with the SimilarityRank-method. The recall values for the top-5 recommendations (recall@5) for the three hybrid ranking methods are displayed in Figure 3. On the x-axis we plotted the weight coefficient $\alpha = [0...1]$ and on the y-axis we plotted the according recall values for the proposed hybrid ranking mechanisms. Obviously, setting α to 1 corresponds to the result of the SimilarityRank method. On the other hand, $\alpha = 0$ leads to the same result as the sole execution of the second ranking method used for the hybrid ranking method. This way, also the base ranking methods can be compared to the hybrid methods as at $\alpha = 0$, simTimeRank corresponds to TimeRank, SimPopularityRank corresponds to PopularityRank and SimRecCountRank corresponds to RecCountRank. The Figure shows that SimRecCountRank performs best for all weight coefficients. The other ranking methods, especially SimTimeRank and SimPopRank suffer from the poor performance of the base ranking methods (TimeRank, PopularityRank). This is due to the fact that both TimeRank and PopularityRank do only consider the global factors time and the overall popularity of hashtags and are not considering the actual content of the tweet itself. Using the recency of the tweet might have a bigger effect when using a long-time dataset as basis for the recommendations. In contrast to the time and popularity-based ranking methods, SimRecCountRank considers the context of the hashtag which leads to a good performance. The context of the hashtag is characterized by both similarity of the input tweet and the tweet containing the hashtag candidate and also the number of occurrences within the most similar messages. The overall best result can be reached using SimRecCountRank with α being set to 0.6.

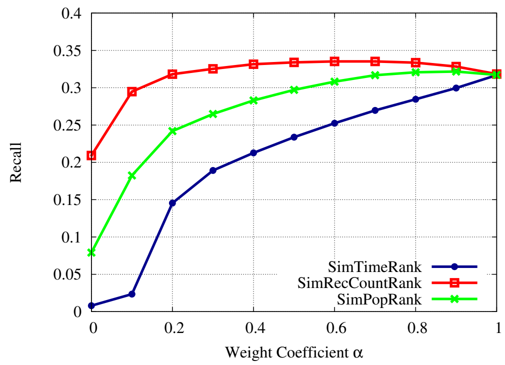

Fig. 3. Recall@5 for Hybrid Ranking Methods

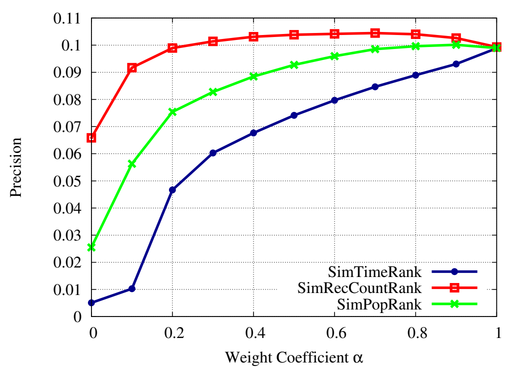

Fig. 4. Precision@5 for Hybrid Ranking Methods

The precision@5 values for the hybrid ranking methods are shown in Figure 4. In general, the precision values reached by our prototype are low. This can be explained by the fact that the number of hashtags used within a tweet is very small. On average, about 1.5 hashtags are used per message. Therefore, evaluating the precision values for e.g. ten recommendations for tweets which do only contain two hashtags naturally leads to very low precision values. Even if the recommendations were 100% correct, still eight other recommended hashtags were not suitable and therefore decrease the precision value. The F-measure of the hybrid ranking methods with $k = 5$ is shown in Figure 5 and underlines the performance of the ranking method SimRecCountRank.

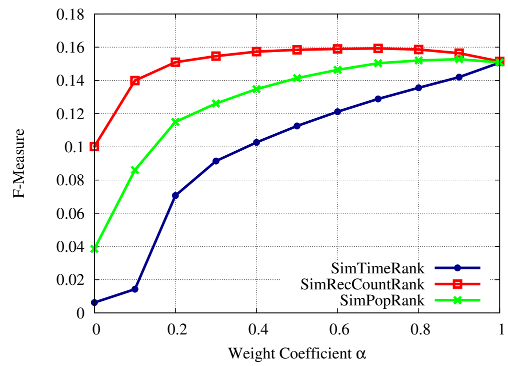

Fig. 5. F-Measure@5 for Hybrid Ranking Methods

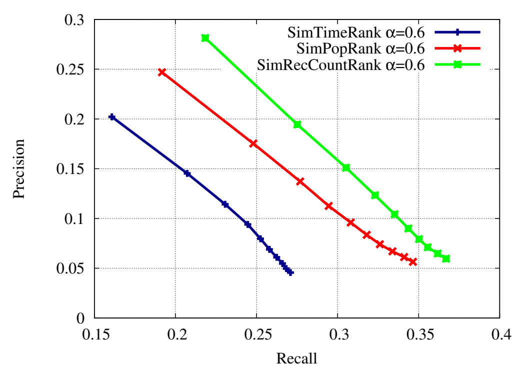

Fig. 6. Precision/Recall Plot for weight α=0.6 and k=[1...10]

In order to further investigate the behavior of the hybrid approaches, we also evaluated the precision/recall values for the described ranking methods. We set the merge coefficient to $\alpha = 0.6$ as this has in general proven to lead to the best results. The resulting recall/precision plot can be seen in Figure 6 where the recall values with $k = 1, 2, ..., 10$ of the corresponding ranking methods are plotted on the x-axis and the precision values are plotted on the y-axis. It turned out that the hybrid SimRecCountRank performed best overall whereas the performance of the other two hybrid ranking methods were rather poor.

4.2 Refinement of Recommendations

In order to show how our recommendation approach performs and how the recommendations are refined with every keystroke during the creation, we compute the recall and precision values of the input tweet at ten different stages during the process of entering a tweet. Therefore, we take the original tweet (without hashtags) and compute the precision and recall values for 10%, 20%, ..., 90%, 100% of the text. The average length of tweets in our datasets are 98 characters without hashtags. Thus, we started the evaluation using an input tweet containing about 10 characters of the original message and evaluated the proposed recommendation algorithms. We proceeded with the recommendation computations until the original length of the tweet without hashtags was reached. The results using a weight α of 0.6 can be seen in Figure 7. It can be seen that constraining the length of an input string directly influences the performance of the ranking methods. The plot shows that the recommendations for a tweet which has only been entered partly, the SimRecCountRank performs significantly better than the other ranking methods. However, it is remarkable that the ranking strategies which take global factors like time or popularity into account performed reasonably well for short input strings. Therefore, we elaborated this fact further and analyzed the behaviour of the different ranking strategies if only 20% of the text were entered. Figure 8 shows the recall values of the different ranking strategies in which the according weight coeffients α are plotted on the x-axis. As the available part of the message is very short, we expected an increasing performance of the ranking methods SimTimeRank and SimPopRank. We also evaluated the different weights of the hybrid ranking methods as shown in Figure 8. Even if the tweet is cut down to 20% of its original length, the SimRecCountRank still performs best – despite the lack of context. This ranking method has proven to be the best performing method regardless of the length of the input tweet.

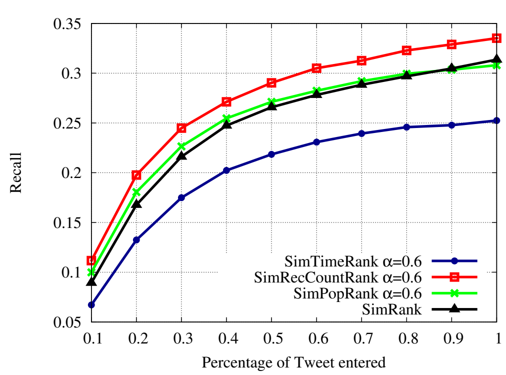

Fig. 7. Development of Recall Values as the User advances in entering the Tweet.

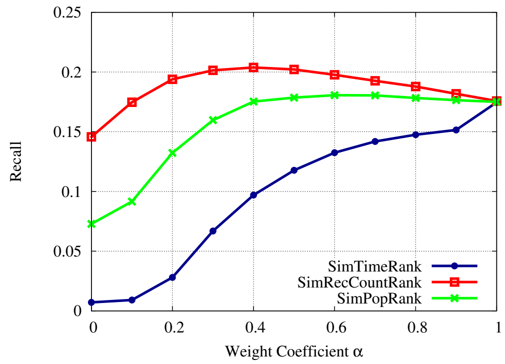

Fig. 8. Recall Values for weight α=0.6 with 20% of the Message as Input

5 Related Work

The recommendation of hashtags within the Twittersphere is closely related to the field of microblogging, tagging in Web 2.0 applications and the field of recommender systems as a whole. Tagging of online resources has become popular with the advent of Web 2.0 paradigms. However, the task of recommending traditional tags differs considerably from recommending hashtags. Our recommendation approach is solely based on 140 characters whereas in traditional tag recommender systems, much more data is taken into consideration for the computation of tags recommendations. Furthermore, tweets, hashtags and trends within the Twittersphere are changing at a fast pace and are very dynamic. New hashtags may evolve around trending topics and therefore the recommendations have to consider this dynamic nature of Twitter.

Sigurbjörnsson *et al.* [23] presented an approach for the recommendation of tags within Flickr which was based on the co-occurrence of tags (also used in [7,15]). Two different tags co-occur if they are both used for the same photo. Based on this information about the co-occurrence of tags for Flickr photos, the authors developed a prototype which is able to recommend hashtags for photos which have been partly tagged. This recommendation is computed by finding those tags which have been used together with the tag the user already specified for a certain photo. These tags are subsequently ranked and recommended to the user. It is important to note that such an approach is not feasible if a photo has not been tagged at all. Partly based on this work, Rae *et al.* [19] proposed a method for Flickr tag recommendations which is based on different contexts of tag usage. Rae distinguishes four different context which are used for the computation of recommendations: (i) the user's previously used tags, (ii) the tags of the user's contacts, (iii) the tags of the users which are members of the same groups as the user and (iv) the collectively most used tags by the whole community. A similar approach has also been facilitated by Garg and Weber in [6]. Furthermore, e.g. on the BibSonomy platform which basically allows its users to add bibliographic entries the users are provided with recommendations for suitable tags annotating these entries [15]. This approach extracts tags which might be suitable for the entry from the title of the entry, the tags previously used for the entry and tags previously used by the current user. Based on these resources, the authors propose different approaches for merging these sets of tags. The resulting set is subsequently recommended to the user. Tag recommendations based on Moviebase data has been presented in [22]. Jäschke *et al.* [11] propose a collaborative filtering approach for the recommendation of tags. The authors therefore construct a graph based on the users, the tags and the tagged entities. Within these graphs, the recommendations are computed and ranked based on a PageRank-like ranking algorithm for folksonomies. Recommendations based on the content of the entity which has to be tagged have been studied in [24]. Additionally, there have been numerous papers concerned with the analysis of the tagging behavior and motivation of users [2, 16].

The social aspects within social online media, such as the Twitter platform, has been analysed heavily throughout the last years. These analysis were concerned with the motivations behind tweeting, like e.g. in [12]. Boyd *et al.* [4] showed how users make use of the retweet function and why users retweet at all. Honeycutt and Hering examined how direct Twitter messages can be used for online collaboration [9]. Recently, the work by Romero *et al.* [21] analyzed how the exposure of Twitter users to hashtags affects their hashtagging behavior and how the use of certain hashtags is spread within the Twittersphere. The authors found that the adoption of hashtags is dependent on the category of the tweet. E.g. hashtags concerned with politics or sports are adopted faster than hashtags concerned with any other topic category. Further analysis of Twitter data and the behavior of Twitter users can be found in [10, 13, 14, 25].

As for the recommendation of items within Twitter or based on Twitter data, there have been numerous approaches dealing with these matters. Hannon *et al.* [8] propose a recommender system which provides users with recommendations for users who might be interesting to follow. Chen *et al.* present an approach aiming at recommending interesting URLs to users [5]. The work by Phelan, McCarthy and Smyth [18] is concerned with the recommendation of news to users. Traditionally, recommender systems are used in e-commerce where users are provided with recommendations for interesting products, like e.g. on the Amazon website. Recommendations are typically computed based on one of the following two approaches: (i) a collaborative filtering [1, 20] approach which is based on finding similar users with a similar behavior for the recommendation of e.g. tags used by these users and (ii) a content-based approach [3, 17] which aims at finding items having the most similar characteristics as the items which have already been used by the user.

However, to the best of our knowledge, there is currently no other approach aiming at the recommendation of tags in microblogging platforms and hashtags for a certain Twitter message.

6 Conclusion

In this paper we presented an approach aiming at the recommendation of hashtags to microblogging users. Such recommendations help the user to (i) use more appropriate hashtags and therefore to homogenize the set of hashtags and (ii) encourage the users to use hashtags as siutable hashtags recommendations are provided. The approach is based on analyzing tweets similar to the tweet the user currently enters and deducing a set of hashtag recommendation candidates from these Twitter messages. We furthermore presented different ranking techniques for these recommendation candidates. The evaluations we conducted showed that our approach is capable of providing users with suitable recommendations for hashtags. The best results were achieved by combining the similarity of messages and the popularity of hashtags in the recommendation candidate set. Future work will include incorporating the social graph of Twitter users into the process of computing recommendations for hashtags to optimize the presented hashtag recommendation approach.

References

1. G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749, 2005.
2. M. Ames and M. Naaman. Why we tag: motivations for annotation in mobile and online media. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '07, pages 971–980, New York, NY, USA, 2007. ACM.
3. M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *Commun. ACM*, 40:66–72, March 1997.
4. D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *hicss*, pages 1–10. IEEE Computer Society, 1899.
5. J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi. Short and tweet: experiments on recommending content from information streams. In *Proceedings of the 28th international conference on Human factors in computing systems*, pages 1185–1194. ACM, 2010.
6. N. Garg and I. Weber. Personalized, interactive tag recommendation for flickr. In *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, pages 67–74, New York, NY, USA, 2008. ACM.
7. W. Gassler, E. Zangerle, and G. Specht. The Snoopy Concept: Fighting Heterogeneity in Semistructured and Collaborative Information Systems by using Recommendations. In *The 2011 International Conference on Collaboration Technologies and Systems (CTS 2011)*, Philadelphia, PE, 05/2011 2011.
8. J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *RecSys '10: Proceedings of the fourth ACM conference on Recommender systems*, pages 199–206, New York, NY, USA, 2010. ACM.
9. C. Honeycutt and S. C. Herring. Beyond Microblogging: Conversation and Collaboration via Twitter. In *HICSS*, pages 1–10. IEEE Computer Society, 2009.
10. B. Huberman, D. Romero, and F. Wu. Social networks that matter: Twitter under the microscope. *First Monday*, 14(1):8, 2009.

11. R. Jaeschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag Recommendations in Folksonomies. In J. Kok, J. Koronacki, R. Lopez de Mantaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *Knowledge Discovery in Databases: PKDD 2007*, volume 4702 of *Lecture Notes in Computer Science*, pages 506–514. Springer Berlin / Heidelberg, 2007.
12. A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM, 2007.
13. B. Krishnamurthy, P. Gill, and M. Arlitt. A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, pages 19–24. ACM, 2008.
14. H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600. ACM, 2010.
15. M. Lipczak and E. Milios. Learning in efficient tag recommendation. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 167–174, New York, NY, USA, 2010. ACM.
16. C. Marlow, M. Naaman, D. Boyd, and M. Davis. HT06, tagging paper, taxonomy, Flickr, academic article, to read. In *Proceedings of the seventeenth conference on Hypertext and hypermedia*, HT '06, pages 31–40, New York, NY, USA, 2006. ACM.
17. M. Pazzani and D. Billsus. Content-Based Recommendation Systems. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 325–341. Springer Berlin / Heidelberg, 2007.
18. O. Phelan, K. McCarthy, and B. Smyth. Using twitter to recommend real-time topical news. In *Proceedings of the third ACM conference on Recommender systems*, pages 385–388. ACM, 2009.
19. A. Rae, B. Sigurbjörnsson, and R. van Zwol. Improving tag recommendation using social networks. In *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO '10, pages 92–99, Paris, France, France, 2010. Le Centre de Hautes Etudes Internationales d'Informatique Documentaire.
20. P. Resnick and H. Varian. Recommender systems. *Communications of the ACM*, 40(3):58, 1997.
21. D. M. Romero, B. Meeder, and J. M. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In S. Srinivasan, K. Ramamritham, A. Kumar, M. P. Ravindra, E. Bertino, and R. Kumar, editors, *WWW*, pages 695–704. ACM, 2011.
22. S. Sen, J. Vig, and J. Riedl. Tagommenders: connecting users to items through tags. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 671–680, New York, NY, USA, 2009. ACM.
23. B. Sigurbjörnsson and R. Van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceeding of the 17th international conference on World Wide Web*, pages 327–336. ACM, 2008.
24. M. Tatu, M. Srikanth, and T. D'Silva. *RSDC'08: Tag Recommendations using Bookmark Content*. Workshop at 18th Europ. Conf. on Machine Learning (ECML'08) / 11th Europ. Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD'08), 2008.
25. S. Ye and S. Wu. Measuring Message Propagation and Social Influence on Twitter. com. In *Social Informatics: Second International Conference, Socinfo 2010, Laxenburg, Austria, October 27-29, 2010, Proceedings*, page 216. Springer-Verlag New York Inc, 2010.