Word Similarity Based Model for Tweet Stream Prospective Notification

Abdelhamid Chellal, Mohand Boughanem, Bernard Dousset

IRIT University of Toulouse UPS, Toulouse, France
{abdelhamid.chellal,mohand.boughanem,bernard.dousset}@irit.fr

Abstract. The prospective notification on tweet streams is a challenge task in which the user wishes to receive timely, relevant, and non-redundant update notification to remain up-to-date. To be effective the system attempts to optimize the aforementioned properties (timeliness, relevance, novelty and redundancy) and find a trade-off between pushing too many and pushing too few tweets. We propose an adaptation of the extended Boolean model based on word similarity to estimate the relevance score of tweets. We take advantage of the word2vec model to capture the similarity between query terms and tweet terms. Experiments on the TREC MB RTF 2015 dataset show that our approach outperforms all considered baselines.

Keywords: Prospective notification, tweet summarization, word2vec.

1 Introduction

User generated content (UGC) in social media streams provides valuable information about what is happening in the world and covers both scheduled and unscheduled events. In many cases, Twitter provides the latest news before traditional media, especially for unscheduled events. Hence, social media streams seem to be the appropriate source of information to fulfill the information need of a user who is looking for updates to be timely pushed for topics of interest. In this context, two different tasks can be defined: retrospective summarization and prospective notification [4]. In the first, documents (tweets) are known in advance while, in the second, the stream is filtered in real time and relevant and non-redundant updates are pushed immediately to the user.

However, unlike traditional data sources, UGC in social media streams is characterized by the volume, velocity and variety of the published information. Indeed, the published posts can vary significantly in terms of quality. These features make prospective notification of social media streams a challenging task.

TREC 2015 Microblog Real-Time Filtering (MB-RTF) [2] and TREC Real-Time Summarization 2016 [1] are two evaluation campaigns. The objective of the task is to identify relevant tweets from the stream and send those updates directly to the user's mobile phone. The main issue in this task is to find a

[1] http://trecrts.github.io/TREC2016-RTS-guidelines.html

trade-off between pushing too many and pushing too few tweets. In the later case, the user may miss important updates and in the former case, the user may be overwhelmed by irrelevant and/or redundant information.

This paper explores a novel approach for prospective notification in tweet streams that pushes in a real time fashion the most salient (relevant and non-redundant) information related to an ongoing event as soon as it occurs in the stream. Knowing that an effective system needs to optimize three properties: The relevance with respect to the topic of interest, the novelty/redundancy and the latency between the publication time and the notification time of selected tweets. To fulfill these requirements, the proposed approach consists of three filters that are adjusted sequentially and in which the decision to select/ignore a tweet is made immediately. The first filter is a simple tweet quality and topicality filter, the second filter is related to the relevance and the third one is for novelty control in order to avoid pushing redundant information to the user.

The main contribution of this paper is the proposition of an adaptation of the Extended Boolean Model (EBM)[5] based on word similarity to estimate the relevance of the incoming tweet with respect to the topic of interest. In addition, instead of using the TF-IDF weighting technique, the query term weight is estimated by taking advantage of the word2vec model [3]. It is estimated through its similarity with the tweet's terms, computed by cosine similarity between word vector generated by the word2vec model. Indeed, the novelty score of the incoming tweet is measured using word overlap with respect to words of tweets already pushed. The defined novelty function avoids a pairwise comparison allowing to reduce the computational complexity. The experiments conducted on the TREC MB RTF 2015 dataset show that our approach outperforms all the baselines.

2 Related Work

Prospective notification in social media streams is the task in which user wishes to receive timely, relevant and non-redundant updates [4]. The TREC MB RTF-2015 official results reveal that runs PKUICSTRunA2 [1] and UWaterlooATDK [6] are the two best performing ones among 37 runs from 14 groups [2]. In the former, the relevance score of tweets is evaluated by using the normalized KL-divergence distance and the decision to select a tweet is based on a predefined threshold set using human intervention. They manually scan the ranked list of top-10 selected tweets of previous day from top to bottom, and the relevance score of the first irrelevant tweet is chosen as a threshold in the next day for the related topic. In UWaterlooATDK run, the relevance score is based on the query term occurrence in the tweet. The threshold is fixed for each day according to the score of the top-10 tweets returned in the previous day. In [7] authors improve the effectiveness of their approach (UWaterlooATDK) by using a daily feedback strategy to estimate the relevance threshold for the next day. However, one can argue that a daily interaction for ongoing feedback judgment might be too onerous in practice. We show in this work that the result reported in [7] is outperformed by the proposed approach, in which the threshold is set adaptively without the use of feedback.

3 Real-Time Tweet Filtering

Our approach acts like a filter with three levels related to the topicality of tweet, its relevance and its novelty respectively. The decision of pushing/ignoring the incoming tweet is made immediately as the tweet occurs. Tweets that pass all these filters are selected as a summary which is denoted by S.

3.1 Tweet quality and topicality filter

The first filter eliminates non-English tweets and those containing less than three tokens. It also drops all incoming tweets that do not contain a predefined number of query words. The incoming tweet T is considered as a candidate tweet if its number of overlapping words with the query title is higher than the minimum of either a predefined constant (K) or the number of words in the query title $min(K, |Q^t|)$. Pilot experiments on TREC MB RTF dataset revealed that the filter $k = 2$ captures about 40% of relevant tweets while the filter $k = 1$ returns 74% of relevant tweets but it also brings up a lot of noise. These results motivated our choice to set $k = 2$.

3.2 Relevance filter

Tweets have a limit length of 140 characters, are noisy and ungrammatical, which implies that the statistical features such as term frequency may be less useful. We believe that the similarity between the tweet words and query words is the key feature. Hence, to evaluate the relevance score of the incoming tweet with respect to the query, we propose (i) to use the extended Boolean model [5] to evaluate the relevance score of a tweet; (ii) to use the similarity score between query words and tweet words to evaluate the weight of query words.

Assume that the query Q (user interest) consists of a title Q^t and description Q^d of the information need. The query title Q^t represents "**AND**ed terms" while Q^d represents "**OR**ed terms". In the Extended Boolean Model, the relevance scores of tweet $T = \{t_1, ..., t_n\}$ to "AND query" Q^t and "OR query" Q^d are estimated respectively as follows:

$$RSV(T, Q^t_{and}) = 1 - \sqrt{\frac{\sum_{q_i^t \in Q^t}(1 - W_T(q_i^t))^2}{|Q^t|}} \quad (1)$$

$$RSV(T, Q^d_{or}) = \sqrt{\frac{\sum_{q_i^d \in Q^d}(W_T(q_i^d))^2}{|Q^d|}} \quad (2)$$

where $W_T(q)$ is the weight of the query term q in the tweet T. $|Q^t|$ and $|Q^d|$ are the length of the title and the description of the query respectively. q stands for the term q_i^t in the query title Q^t or the term q_i^d in the query description Q^d.

Instead of using the TF-IDF like weighting schema, we propose to estimate the weight $W_T(q)$ by evaluating the similarity between the query term q and all the terms of tweet T as follows:

$$W_T(q) = \max_{t_i \in T}[w2vsim(t_i, q)] \quad (3)$$

where $w2vsim(t_i, q)$ is the similarity between tweet word t_i and query word q. We propose to represent terms using their word2vec [3] representation and the similarity between two terms is measured by cosine similarity between their word2vec vectors. The intuition behind this proposition is that tweets that have words sharing many contexts with the query words will be more relevant. The main advantage of using word2vec model to estimate the similarity between two words is that a query word which does not appear in a tweet but shares many contexts with the tweet words will get a weight different from 0. Indeed, the relevance score of an incoming tweet is evaluated at the time the new tweet arrives, independently of tweets previously seen in the stream and without the need for indexing the tweet stream. The word vectors used in our experiments to estimate the word similarity were generated using tweets crawled during 9 days before evaluation period.

With $RSV(T, Q^t_{and})$ and $RSV(T, Q^d_{or})$, we got two relevance scores for tweet T regarding the title and the description of the query, respectively. The final relevance score of tweet T is measured by combining the aforementioned scores linearly with title terms having greater weight than description terms as follows:

$$RSV(T, Q) = \lambda \times RSV(T, QT_{and}) + (1 - \lambda) \times RSV(T, QD_{or}) \tag{4}$$

where $\lambda \in [0, 1]$ is a parameter determining the trade-off between the query title's words and the description's words. Based on pilot experiments where λ was varied from 0 to 1 in increments of 0.1, the weight λ was set to 0.8 .

A tweet passes the relevance filter if its score is above a certain threshold. This threshold is estimated at the decision time based on the previous values. Our thresholding strategy is to consider the average of the previously seen values of the relevance score. However, we do not lower the threshold under a global minimum threshold GT. Hence the relevance threshold is defined by $max(GT, avg(RSV(T, Q))$

3.3 Novelty filter

The intuitive way to estimate the novelty of an incoming tweet is to conduct a pairwise comparison with previously seen tweets in the stream using a standard similarity function such as cosine similarity or KL-divergence. Due to the limited length of tweets, meaningful words rarely occur more than once which implies that aforementioned similarity functions are less useful for evaluating the distance between two tweets. Indeed, a pairwise comparison does not fit a real-time scenario. For these reasons, we propose to merge all tweets already selected in the summary into a "summary word set "and evaluate the novelty score using the number of overlapping words between the incoming tweet and summary word set. Assume that SW is the set of words that occur in current summary, then the novelty score of the incoming tweet T is evaluated as follows:

$$NS(T, SW) = 1 - \frac{|SW \cap T|}{|T|} \tag{5}$$

Tweets with novelty score less than 0.6 were discarded. We set the novelty threshold experimentally using TREC MB RTF 2015 dataset.

4 Experimental evaluation and results

Experiments were conducted on the TREC 2015 Microblog Real Time Filtering (MB RTF) track dataset by using replay mechanism over tweets captured during the evaluation period. This collection was generated by each participant independently using Twitter's streaming API during the 10 days of the evaluation period (20 July to 29 July 2015). In our experiments, we focus on the scenario "Push notifications" which corresponds to a real-time task and where a maximum of 10 tweets per day per topic are returned. The organizers defined two evaluation measures that consider both the relevance and the time at which they were pushed [2]. The primary metric is the expected latency-discounted gain (ELG) in which a latency penalty is applied. The second metric is the normalized cumulative gain (nCG). These two metrics are defined as follows:

$$ELG(S) = \frac{1}{N} \times \sum_{T \in S} G(T) \times max(0, (100 - delay)/100) \tag{6}$$

$$nCG(S) = \frac{1}{Z} \times \sum_{T \in S} G(T) \tag{7}$$

where S is the generated summary, N is the number of returned tweets and Z is the maximum possible gain (given the 10 tweet per day limit). The delay is the latency (in minutes) between the tweet creation time and the time the system decides to push it. G(T) is the gain of each tweet which is set as follows: irrelevant tweets receive a gain of 0, relevant tweets receive a gain of 0.5 and highly relevant tweets receive a gain of 1.0.

Thresholding impact: To better understand the impact of the threshold used in the relevance filter, we plot in Figure 1 the effectiveness of our system in terms of ELG and nCG across a range of global threshold values. The baseline in this experiment is the empty run and the oracle run which represents the run where only relevant tweets that pass the first filter are selected. Figure 1 shows that the best results in terms of both metrics are achieved with global threshold GT = 0.5. The threshold controls the number of pushed tweets. Note that the empty run is a challenging baseline that many systems in TREC 2015 failed to beat. For some days, no relevant tweets occur and the system should push nothing in this case. Also, the comparison between the best results and the oracle run reveals that more improvements can be achieved through better filtering and threshold setting.

Comparative evaluation with state-of-the-art approaches: In this section, we compare our approach (denoted by WSEBM for Word Similarity EBM) against the two high-performing official results from the TREC MB-RTF 2015 PKUICSTRunA2 [1] and UWaterlooATDK [6] and against the approach described in [7] in which they improve their results obtained in TREC 2015. In addition, in order to evaluate the impact of using word similarity as weighting technique, we compare our method with standard EBM. This baseline is based on the proposed functions (4) in which we consider the query term number of occurrences in a tweet as the term's weight. Table 1 reports the results in terms of ELG and nCG. As shown in this table, the WSEBM outperforms all baselines

Fig. 1. ELG and nCG for different thresholds, the oracle run and the empty run.

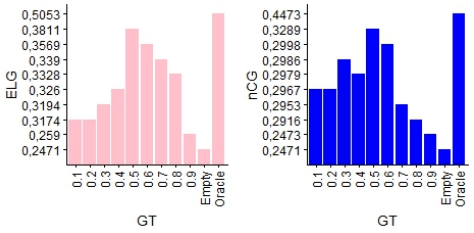

Table 1. Comparative evaluation with state-of-the-art.

Method	ELG	nCG	%ELG
WSEBM	**0.3811**	**0.3289**	
EBM	0.2583	0.2544	+32.22%
Tan et al[7]	0.3678	-	+3.48%
TREC MB RTF 2015 official Results			
PKUICSTRunA2	0.3175	0.3127	+16.68%
UWaterlooATDK	0.3150	0.2679	+17.34%

Note. % indicates improvements in terms of ELG.

overall metrics. We found performance improvements up to ELG values of about 16% for the best run in TREC MB 2015 task and of about 3.4% for the approach based on feedback strategy to set the relevance threshold.

5 Conclusion

In this paper, we introduced a new approach for prospective notification in which we show that word similarity matching and simple thresholding strategy achieve good results in terms of expected and cumulative gain. The use of the semantic word relationships improves the efficiency of the relevance filter. The proposed relevance function enables the use of simple threshold across all topics. The results showed that better results can be achieved if the threshold is appropriately set. In future work, we plan to leverage social signals to filter incoming tweets.

References

1. Fan, F., Fei, Y., Lv, C., Yao, L., Yang, J., Zhao, D.: Pkuicst at trec 2015 microblog track: Query-biased adaptive filtering in real-time microblog stream. In: Text REtrieval Conference, TREC, Gaithersburg, USA, November 17-20 (2015)
2. Lin, J., Efron, M., Wang, Y., Sherman, G., McCreadie, R., Sakai, T.: Overview of the trec 2015 microblog track. In: Text REtrieval Conference, TREC, Gaithersburg, USA, November 17-20 (2015)
3. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
4. Qian, X., Lin, J., Roegiest, A.: Interleaved evaluation for retrospective summarization and prospective notification on document streams. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 175–184. SIGIR '16 (2016)
5. Salton, G., Fox, E.A., Wu, H.: Extended boolean information retrieval. Commun. ACM 26(11), 1022–1036 (Nov 1983), http://doi.acm.org/10.1145/182.358466
6. Tan, L., Roegiest, A., Clarke, C.L.: University of waterloo at trec 2015 microblog track. In: Text REtrieval Conference, TREC, Gaithersburg, USA, Nov 17-20 (2015)
7. Tan, L., Roegiest, A., Clarke, C.L., Lin, J.: Simple dynamic emission strategies for microblog filtering. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1009–1012. SIGIR '16 (2016)