Summarizing tweet in real-time by filtering quality, relevant and non redundant tweets

Bilel Moulahi[1], Lamjed Ben Jabeur[2], Rafik Abbes[3], and Lynda Tamine[3]

[1] LIRMM, University of Montpellier, CNRS, 34095 Montpellier Cedex 5
moulahi@lirmm.fr
[2] P3 Group, 3 Boulevard Henri Ziegler, 31700 Blagnac,
Lamjed.BenJabeur@p3-group.fr,
[3] IRIT, Université de Toulouse UPS-IRIT, 118 route de Narbonne F- 31062 Toulouse cedex 9
{abbes,tamine}@irit.fr

Abstract. This paper presents the participation of LIRMM laboratory (University of Montpellier), P3 Group and IRIT laboratory (University of Toulouse) to the Real Time Summarization track of TREC 2017. We extend our previous approach [1] for real-time filtering of tweet stream that aims to identify quality, relevant and non-redundant tweets to be pushed to the user at real-time. We describe in this paper the proposed approach and we discuss official obtained results.

Keywords: real-time, social media, word similarity, filtering

1 Introduction

Social media streams provide real time updates that cover scheduled and unscheduled events which makes them a valuable source of information for user who wishes to receive timely notification to keep up-to-date on topic of interest. Indeed of the volume and the redundancy of the posted information in the social media stream, one of the main challenge consists in the fact that to be effective, notifications must achieve a balance between pushing too much and pushing too little. Push too little and the user misses important updates; push too much and the user is overwhelmed by irrelevant/redundant information [2]. Although several models have been proposed in the context of ad-hoc tweet search [3, 4], the task of prospective notification in tweet stream, which is proposed by the real time summarization Track of TREC 2017, is still under-explored.

The Microblog real time summarization Track aim at monitoring the social media data-stream in order to push tweets to users with respect to their topical interest-based profile. One main assumption yields in the TREC guidelines is that notifications and digests might enable the user to keep up-to-date on the topic of interest. In this aim, the track is split into two main scenarios:

1. The Scenario *A*, called "*Push notifications*", consists in an instantly tweet notification assuming a short time period between the tweet publication and the tweet pushing. Participating system are allowed to push up to 10 notifications per day per interest profile.
2. The scenario *B*, called "*Periodic email digest*" consist on daily selecting up to 100 tweets for each interest profile to be send to user via email. It's required that tweets should be relevant and novel but timeliness is not important.

In this paper, we investigate a filtering model aiming at retrieving tweets in a real-time *fashion* with respect to the push and digest scenarios. The filtering model decompose the filtering task to several sub-tasks in accordance to the summarization constraints. The final filtering score is aggregated as the product of scores obtained by sub-filtering functions.

This paper is organized as follows. We introduce in section 2 our filtering model for real-time tweet summarization. We present in section 3 official results. Section 3, concludes the paper and announces future works.

2 Streaming filter model for real-time summarization

Real-time summarization of tweets consist in selecting from a continuous stream of tweets $T =< t_1, t_2, \ldots, t_n >$ the set of relevant ones with respect of the tracked topic q. The result summary $S =< t_1, t_2, \ldots, t_m >$ must not include redundant tweets and respect a length constraint in terms of the number of tweets. The summarization problem is considered as a filtering task where filtering function $F(t_i)$ is applied to the incoming tweet ti in order to decide if the tweet must be included in summary $F(t_i) = 1$ or neglected $F(t_i) = 0$. We propose here a filtering model that combines several summarization constraints that must be verified namely the tweet quality, topical relevance and information redundancy.

2.1 Streaming filter model

Tweets to be included in the summary must respect several constraints for instance the summary length, the topical relevance and the non-redundancy of information with regards to past included tweets. Hence, we propose to decompose the filtering task into sub-filtering tasks where each one verify that the tweet respect a particular constraint. A set of sub-filtering functions $F_k(t_i)$ is defined in accordance to the k^{th} summarization constraints with $F_k(t_i) = 1$ allows to include tweet ti in the summary or $F_k(t_i) = 0$ otherwise. The global filtering function $F(t_i) = 1$ is computed as the product these functions, requiring that tweet t_i must verify all constraints. $F(t_i)$ is computed as the following.

$$F(t_i) = \prod_{\forall k} F_k(t_i) \tag{1}$$

Table 1 lists summarization contains that we suggest to consider for summarization real-time stream. Further constraints could be add this list in order to satisfy advance summarization constraints and users preferences. A detailed description of computation of each filtering function is defined in the next section.

Table 1. Summary constraints and respective filtering functions

Function	Constraint	Description
$F_0(t_i)$	Summary length	Summary must be limited to l tweets
$F_1(t_i)$	Language preference	Tweet t_i must be in user's language
$F_2(t_i)$	Lexical restriction	Tweet t_i must not include "swearing words"
$F_3(t_i)$	URL presence	Tweet t_i must include an URL
$F_4(t_i)$	Topical relevance	Tweet t_i must be relevant to tracked topic
$F_5(t_i)$	Non redundancy	Tweet t_i must not be redundant

2.2 Computing filtering scores

We details in the section the computation of filtering functions introduced in table 1.

Summary length. The length constraint suggest that the length of the summary must not overpass l. As proposed in summarization scenarios A and B , the length constraints is defined for a limited time window (i.e. day). The length constraint set to $l = 10$ and $l = 100$ respectively to scenario A and scenario B. Let θ_i be the timestamp of tweet t_i and $d(\theta_i)$ the day in witch t_i is published. The length of the current daily summary is defined by the subset of tweets $S_{d(\theta_i)} = \{t_j \in S | d(\theta_j) = d(\theta_i)\}$. Accordingly, we define the filtering function $F_0(t_i)$ in respect of summary length as the following:

$$F_0(t_i) = \begin{cases} 1, & |S_{d(\theta_i)}| < l \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Language preference. It is interesting to push tweets in user's language preferences. Tweets in other languages are useless they are translated to preferred language. For this aim we propose language filtering score $F_1(t_i)$ as the following:

$$F_1(t_i) = \begin{cases} 1, & \text{lang}(t_i) \in G \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

With $\text{lang}(t_i)$ is the language of the tweet and G the set defining language preferences of the user. In this experiment we set $G = \{en\}$ since only English tweet are taken into consideration.

Lexical restriction. In order to ensure a high quality of tweet, we propose to apply a lexical filter that particularly eliminate tweet containing "swearing words". Let $L = \{w_1, w_2, ..., w_p\}$ be the lexicon of undesirable words. The lexical filtering score $F_2(t_i)$ is computed as the following:

$$F_2(t_i) = \begin{cases} 1, & |t_i \cup L| = 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The lexical filter may be extended to include other types of lexicon in addition to "swearing words" such as "power words" (e.g. "free", "easy" and "best") for catching attention and emotionally affect reader.

URL presence. Experiments results on previous TREC microblog track show that presence of URL is a good indicator of the relevance of the tweet. Accordingly we propose a strict filtering constraint that suggest to filter out tweet that do not contains any URL.

$$F_3(t_i) = \begin{cases} 1, & \text{urls}(t_i) \geq 1 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

With $\text{urls}(t_i)$ is the number of URLS on the tweet.

Topical relevance. Notified tweet may be relevant with regards to interest profile. In order to determine the relevance of the tweet we adopt a strict policy suggesting that a significant number of terms from tracking topic must be included in the tweet. In fact, tracking topics are different from regular search query in terms motivation since user have an exact idea about what she is looking and carefully choose a tracking query that target his need. Hence we propose that tweet must contains at least α terms otherwise or all terms of tracking topic q if $|q| < \alpha$. We refer to the topic topic q to interest profile title as described track guideline. The topical filtering score $F_4(t_i)$is defined as the following:

$$F_4(t_i) = \begin{cases} 1, & |q \cup t_i| \geq min(\alpha, |q|) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

with $|q \cup t_i|$ is the number of distinct common terms between the topic and the tweet. We propose to set up parameter in this experimentation to arbitrary value $\alpha = 3$. This value may be adjusted through experiments.

Non redundancy. Pushed tweets to notification may be not redundant and deliver a new information to user at each time. Otherwise, redundant notifications may cause user fatigue regardless if there are relevant or not. For this aim, we propose to apply a redundancy filter that eliminated tweet containing repeated information to what have been previously notified. In particular, we propose to check the similarity of incoming tweet with recently pushed ones across the past time window Δt.

Let $Q =< t_1, t_1, \ldots, t_q >$ a timed-queue of previously pushed tweet where each tweet t_i is characterized with a timestamp θ_i. All tweets in Q belong to the last window Δt verifying so the condition $\theta_i <= \theta_{now} - \Delta t$, $\forall t_i \in Q$ with θ_{now} is the actual time. We propose to compute a redundancy score of each incoming tweet as the maximum score of similarity to tweets in queue Q. In this context, we define the similarity between two tweets t_i and t_j as the proportion of common terms while taking into account the length of reference tweet ti.This score note $r(t_i, Q)$ is computed as follows.

$$r(t_i, Q) = \underset{t_j \in Q}{\operatorname{argmax}} \frac{|t_i \cup t_j|}{|t_i|} \tag{7}$$

With $|t_i$— is distinct terms in t_i and $|t_i \cup t_j|$ is the number of distinct common terms between t_i and t_j . We note the that normalizing the number of common terms over the length of incoming tweet ti in the similarity quotient $\frac{|t_i \cup t_j|}{|t_i|}$ instead of using the minimum length of two tweets as in overlap coefficient equation $\frac{|t_i \cup t_j|}{min(|t_i|,|t_j|)}$ reduces the fallacy of too short tweets in queue Q in terms of redundancy towards longer new informative tweets.

We compare the redundancy score $r(t_i, Q)$ of t_i to fixed threshold . Tweet t_i will filter out it score overpass β. The filtering redundancy score $F_5(t_i)$ is defined as follows.

$$F_5(t_i) = \begin{cases} 1, & r(t_i, Q) < \beta \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

In this experiments we set up β parameter to arbitrary value $\beta = 0,6$ which represent a strict threshold for eliminating redundant tweet. The value of could be configured experimentally as the average overlap similarity between the relevant tweets of each topic from past released datasets of TREC microblogs. Besides, we set up the time window Δt parameter of the timed-queue Q to arbitrary value of $\Delta t = 1 day$ requiring that user must not receive a redundant tweet in the same day. This value is accepted as most of news as edited daily. Increasing this time window to several days may cause over-filtering of novel tweets that may use partial vocabulary in past ones put to announce a new information.

2.3 Results

As our filtering model is compatible for both scenario A and B, we submitted the same summaries for both of them.For scenario A, tweet are pushed in real time to the evaluation API. The same tweets are used for the make our run for scenario B while raking them based on their recency. We notice that we are limited to 10 tweet per daily digest summary for scenario B although 100 tweets are allowed. The discuss in what follows obtained results for scenario A and scenario B.

scenario A. Our system is able to deliver more relevant tweets compared to the median results. Precision values show that our system slightly overpass the median scores with a value of 0.23 compared to 0.19, that is given by our first run. We notice that the values for all runs are considerably low and this is due to the values of the thresholds that we have fixed to filter irrelevant tweets.

We note that there two versions of expectation gain (EG1 and EGO as well and nCG1 and nCG0. In contrast to EG1 and nCG1, the EGO and nCG0 ignore silent days where no relevant document is published so participating systems receive equal gain. As shown in Table 2, our runs overpass the values of the median scores for the four measures EG1, EGO, nCG1 and nCG0. This highlight that our filtering model is more effective for summarization in terms of the relevance of included tweets as well as recognition that there are no relevant tweets to push.

Table 2. Official results of our system (runs advanse_lirmm-RunX) and median scores for Scenario A.

	EG-1	EG-p	nCG-1	nCG-p	mean latency
advanse_lirmm-Run1	0.2352	0.2686	0.2501	0.2835	38388.7
advanse_lirmm-Run2	0.2327	0.2653	0.2402	0.2728	37569.8
advanse_lirmm-Run3	0.2298	0.2626	0.2498	0.2825	38112.4
Median scores	0.1951	0.2194	0.1826	0.2095	–

scenario B. Table 3 Shows nDCG1 and nDCG0 values obtained by our system. We remember that these measures are computed as the average of nDCG@10 scores for each day and topic. In contrast to nDCG1, the nDCG0 discard silent days so participating systems receive equal gain. As shown in the table 3. The performances of our system overpass the averaged topic median score for all participants for nDCG1. However, this improvement do not imply stable effectiveness of our model due the low performances for nDCG0. A further analysis on our submitted run show that the good performances of obtained by our system for nDCG1 measure are explained due to its timid behavior in contrast of verbose systems continuously sending tweets every day. Beside, results for nDCG0 which also evaluate tweet ranking process show our ranking strategy that suggest to order tweets according have limited performance.

Table 3. Official results of our system (runs adv_lirmm-RunX) and the median scores for Scenario B.

	nDCG@10-1	nDCG@10-p
adv_lirmm-Run1	0.2289	0.2669
adv_lirmm-Run2	0.2227	0.2601
adv_lirmm-Run3	0.2285	0.2656
Median scores	0.1865	0.2194

3 Conclusion and future work

We presented in this paper three different approaches for real time summarization of tweet stream. The proposed approach compute either a relevance score or filtering score which allow to determine if a new tweet should be include . For all these approaches, we underline that further experiments are needed, more particularly in the parameter tuning steps. However, we believe that results are quite promising and could give interesting insights in the future in terms of real-time tweet indexing and retrieval, which are important components in the information access within data-streams.

References

1. Bilel Moulahi, Lamjed Ben Jabeur, Abdelhamid Chellal, Thomas Palmer, Lynda Tamine, Mohand Boughanem, Karen Pinel-Sauvagnat, and Gilles Hubert. Irit at trec real time summarization 2016. In *Proceedings of Int. Conf. TREC'16*, 2016.
2. Charles L. A. Clarke Jimmy Lin Luchen Tan, Adam Roegiest. Simple dynamic emission strategies for microblog filtering. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, 2016.
3. Ian Soboroff, Iadh Ounis, J Lin, and I Soboroff. Overview of the trec-2012 microblog track. In *Proceedings of TREC*, volume 2012, 2012.

4. Jimmy Lin and Miles Efron. Overview of the trec-2013 microblog track. In *Proceedings of TREC*, volume 2013, 2013.