GPLSI at TREC 2018 RTS Track

Javi Fernández*, Fernando Llopis, Yoan Gutiérrez,
Patricio Martínez-Barco, José M. Gómez, Rafael Muñoz
{javifm,llopis,ygutierrez,patricio,jmgomez,rafael}@dlsi.ua.es

Department of Software and Computing Systems
University of Alicante

Abstract

In this paper we present our contribution for the *TREC 2018 Real-Time Summarization* track. This task contains two scenarios: *push notifications*, and *email digest*. We participated in both, submitting three runs on each one. Our main goal was to evaluate the effectiveness the techniques employed in *Social Analytics*, a reputation analysis platform, which finds relevant tweets for specific topics. Here, we describe these techniques, and discuss the results obtained.

1 Introduction

Every second, new information is published on the Web 2.0. Part of the information extracted from that content can be very useful. However, in some cases, it can only be useful if it is obtained in real-time. For example, a political party may want to know what people think about their decisions and, in case of negative opinions, they may want to act in consequence as soon as possible. In cases like this, systems that analyze information in real-time are gaining interest. The *Real-Time Summarization*[1] (RTS) track [4] of the *Text Retrieval Conference*[2] (TREC) focuses on this type of information need.

The task contains two scenarios: *push notifications* (scenario A) and *email digest* (scenario B). Both scenarios require to identify relevant tweets from *Twitter* respect to a list of topics of interest. The main difference is that, in the first scenario, the identification must be performed in real-time and, in the second scenario, a ranking with the most relevant tweets must be created daily [4].

*Corresponding author
[1]`https://trecrts.github.io`
[2]`https://trec.nist.gov`

In this paper we describe the techniques employed to identify relevant texts, among which we can highlight the use of *sentiment analysis* to help this task. The detailed approaches are described in Section 3. This approaches are based on the ones developed for our application *Social Analytics*, a reputation analysis platform, which is briefly described in Section 2. Subsequently, in Section 4 we show the assessment of our model in the competition. Finally, the conclusions and future work are presented in Section 5.

2 Social Analytics

Social Analytics[3] is a web application designed to monitor the reputation of some specific entities (people, products, organizations, etc.) in real-time [2, 3]. These entities are predefined by the users of the application before starting the monitoring. To acomplish this, the data from social networks (*Twitter*[4] and *Instagram*[5] at this moment) is downloaded, analyzed, indexed, and made accessible to the users as soon as it is published. This information is visually offered to the users in different views (see Figure 1), among which we can highlight:

- Number of tweets
- Entity reputation (numeric value)
- Entity competitors reputation and ranking
- Evolution of the reputation in a period of time
- Evolution of the number of tweets in a period of time
- Number of tweets by polarity
- Number of tweets by emotion
- Most employed words and hashtags
- Places where most of the tweets were published

We believe that our application fits perfectly in the focus of the RTS track. In our application, we also have to decide which tweets are relevant to the entities specified by the user in real-time. This is the reason why the techniques used in this task and the ones we employed in our application are very similar. In the following sections we will explain the techniques employed in detail.

[3]`https://socialanalytics.gplsi.es`
[4]`https://twitter.com`
[5]`https://instagram.com`

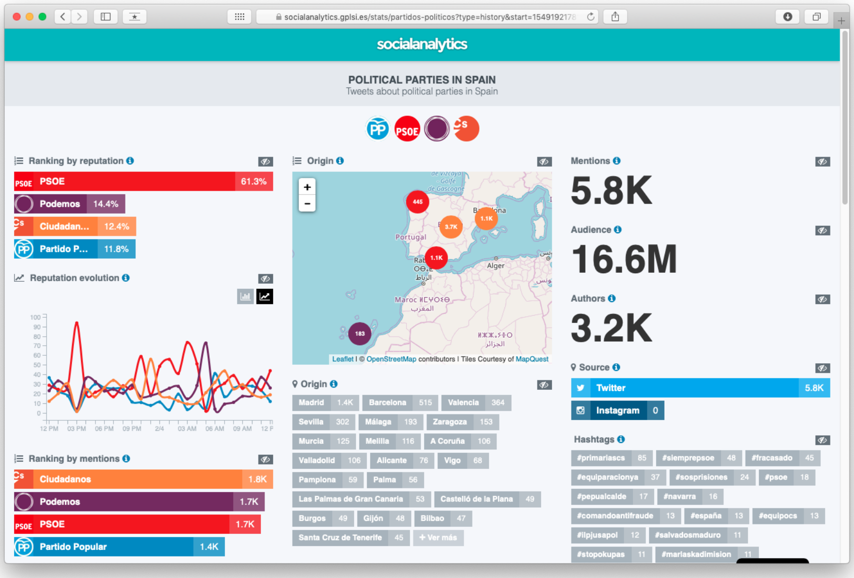

Figure 1: Social Analytics dashboard screenshot

3 Methodology

Although the goal of each scenario is different (*push notifications* and *email digest*), both require to identify relevant tweets respect to the list of topics of interest provided. Section 3.1 shows the techniques employed for this purpose. The subsequent sections 3.2 and 3.3 describe the approaches for scenarios A and B respectively.

It is worth mentioning that all of our approaches are **Automatic Runs**. This is, they operate without human intervention before or during the evaluation period; we do not perform any kind of query expansion before or during the evaluation period; and we do not employ any training data from previous tracks. All the approaches proposed in this work are based on rules designed before the evaluation period started.

3.1 Tweet filtering

In both scenarios, we used the same approach as first filter for all the tweets fetched. This stage has two goals: identify relevant content, and identify redundant content.

Identification of relevant content

In this stage, a tweet is considered as relevant if it satisfies the following criteria:

- The tweet contains all the terms of the topic title.
- The tweet adds information to the topic title.

We consider that a tweet adds information to the topic title if it contains more terms than the topic title. More specifically for the experiments submitted, the tweets must have at least twice as many terms as the topic title to be considered as relevant.

Identification of redundant content

All the tweets identified as relevant are indexed with the *Lucene*[6], a free and open-source information retrieval software library, in order to have a searchable tweet history. We consider a content as redundant if it satisfies at least one of the following criteria:

- The new content contains a URL, and that URL has been mentioned in a previous tweet.
- A high percentage of the terms of the tweet appear in a previous tweet.

In the experiments submitted, the percentage chosen for the second criterion was the 90% of the terms of the new tweet. At this point, we keep only the relevant and non-redundant tweets. This will be our baseline approach.

3.2 Scenario A. Push Notifications

The goal of this task is to identify relevant content and push it to the TREC RTS evaluation broker using its REST API[7]. In this scenario, we want to test the usefulness of *sentiment analysis* in relevance identification tasks. For this purpose, we submitted three runs: one that does not use sentiment analysis (as baseline), and two that use sentiment analysis in different ways to help in the decision of identifying relevant tweets. These are those approaches:

[6] `http://lucene.apache.org`
[7] `https://github.com/trecrts/trecrts-eval/tree/master/trecrts-server`

- **Run A1**. This run only filters the tweets with the techniques mentioned in Section 3.1. This is our baseline.
- **Run A2**. This run filters the tweets with the techniques mentioned in Section 3.1, but also removes those tweets classified as *objective* (or *non-subjective*). In other words, in this run we only push the opinionated tweets, to test if subjective information is more relevant in this scenario.
- **Run A3**. This run filters the tweets with the techniques mentioned in Section 3.1, but also removes those tweets whose subjectivity does not match the subjectivity of the topic. For example, if the topic is classified as *subjective*, we remove the *objective* tweets, and if the topic is classified as *objective*, we remove the *subjective* tweets for that topic. In this run we hypothesize that, if the topic has some degree of subjectivity, relevant tweets must also be subjective.

To detect the polarity of the messages, we used a hybrid approach described in [1, 3], which, in summary, uses *skipgrams* as information units, creates a sentiment lexicon from a labeled dataset, and builds a classifier using that lexicon and machine learning techniques. This approach is currently being used in our application *Social Analytics*.

3.3 Scenario B: Email Digest

The goal of this task is to generate a summary with the most relevant tweets of the day for each topic, selected automatically by the system. In this scenario, there should be no repeated tweets or tweets with the same information. So again, two tasks must be performed: select the relevant tweets to a topic, and discriminate those that contain similar information. The model we have used in this scenario has this steps:

1. Calculate the relevance of the tweets with respect to the question.
2. Sort the tweets by this relevance score.
3. Remove tweets with similar information.
4. Generate a list with the not discarded tweets.

Again, three runs were submitted for this scenario. In summary, a score is assigned to each filtered tweet, and tweets are sorted on the basis of that score:

1. **Run B1**. This run is made up of the 100 most relevant tweets in the system using the cosine model. This is our baseline for scenario B.
2. **Run B2**. This is a dictionary-based model. Each word has a weight that depends on the number of times it appears in the list of tweets of the day (using the cosine model). Each tweet is compared to the rest of the tweets identified as relevant. In this comparison, the weights of the words that

appear in both tweets are added, and the weights of the words that only appear in one are subtracted. If the sum is less than zero, the tweet is discarded.

3. **Run B3**. This process is similar to the previous one, but it is assumed that the weight of each word is 1.

4 Results

The results obtained by our system are slightly above the median of all the participants (see Tables 1 and 2).

	Strict-P	Lenient-P	# Relevant
Median	.4791	.4829	-
A1	**.5546**	**.5615**	4011
A2	.4860	.4921	2416
A3	.5328	.5382	2844

Table 1: Scenario A results

For scenario A, the run that obtained the best results is our base case **A1**. However, run **A3** reduces the number of tweets pushed by 40%, maintaining almost the same precision as the run A1. So, this approach can be employed when a more aggressive filtering is needed.

	nDCG-p	nDCG-1
Median	.8287	.7531
B1	.8378	**.7561**
B2	.8379	**.7561**
B3	**.8387**	.7554

Table 2: Scenario B results

For scenario B, the run that obtained the best results is the run **B3**, but the improvements with respect to the baseline are very small. However, the tweet filtering performed in combination with the use of the cosine model to sort the tweets, provide promising results.

5 Conclusions and Future Work

This was the first participation of the *GPLSI* of the *University of Alicante* in the RTS task. Our main goal was to evaluate the performance of our application *Social Analytics*. We also participated in scenario B with the aim of starting the tests of our system of summaries obtained from the relevant tweets.

We did not want to use data from previous editions given the interest we had in validating the effectiveness of *Social Analytics*. *Social Analytics* is a system capable of locating tweets relevant to a specific question or topic. In addition, it carries out a polarity study and analysis of feelings. Within the experiments that we have carried out, we had as main goal to test if that polarity and the detection of sentiments in the topics and tweets could help to select the most relevant tweets. The conclusions obtained have been that, although they do not improve the base model, they do allow a reasonable reduction of the tweets selected without a significant loss of overall effectiveness.

We have also tested the basic systems of summary management that the *Social Analytics* system includes, to discard repeated tweets or tweets with the same content. These basic systems have allowed us to improve the base case, although with a little significant improvement, which leads us to think that we must add some improvements in these models.

In spite of this, we consider the results to be satisfactory, as they are above the average for the rest of the participants. We know that this year will probably be the last edition of the RTS task, but we hope that new tasks with similar objectives will be created.

Acknowledgements

This research work has been partially founded by the University of Alicante and Generalitat Valenciana, through the projects:

PROMETEU/2018/089

PINGVALUE3-18Y

SocialUniv 2.0 (ENCARGOINTERNOOMNI-1).

References

[1] Fernández, J., Gutiérrez, Y., Gómez, J. M., Martínez-Barco, P., Montoyo, A., and Munoz, R. Sentiment Analysis of Spanish Tweets using a Ranking Algorithm and Skipgrams.

[2] Fernández, J., Llopis, F., Gutiérrez, Y., Martinez-Barco, P., and Diez, A. Opinion mining in social networks versus electoral polls. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017* (2017), pp. 231–237.

[3] Fernández, J., Llopis, F., Martínez-Barco, P., Gutiérrez, Y., and Díez, Á. Analizando opiniones en las redes sociales. *Procesamiento del Lenguaje Natural 58* (2017), 141–148.

[4] Lin, J., Mohammed, S., Sequiera, R., Tan, L., Ghelani, N., Abualsaud, M., McCreadie, R., Milajevs, D., and Voorhees, E. Overview of the TREC 2017 Real-Time Summarization Track. In *The Twenty-Sixth Text REtrieval Conference Proceedings (TREC 2017).*