SINAI at TREC 2018: Experiments in Incident Streams

Miguel Ángel García-Cumbreras
CEATIC, Universidad de Jaén
Jaén, Spain
magc@ujaen.es

Manuel Carlos Díaz-Galiano
CEATIC, Universidad de Jaén
Jaén, Spain
mcdiaz@ujaen.es

Manuel García-Vega
CEATIC, Universidad de Jaén
Jaén, Spain
mgarcia@ujaen.es

Salud María Jiménez-Zafra
CEATIC, Universidad de Jaén
Jaén, Spain
sjzafra@ujaen.es

ABSTRACT

This paper describes the system architecture of the University of Jaén - SINAI team's for the TREC 2018 Incident Streams Track. The goal of the challenge is to automatically process social media streams during emergency situations with the aim of categorizing information and aid requests made on social media for emergency service operators. We explored four alternatives: baseline experimentation, WordNet synonyms, spelling correction and word embeddings. All of them use Support Vector Machine (SVM) as machine learning method. Our experiments reveal that the last approach leads to improve the baseline result.

KEYWORDS

Social Media streams, Incidents, NLP

1 INTRODUCTION

Social media sites (e.g., Twitter, Facebook, Instagram) have emerged as powerful means of communication for people who want to share information on a wide variety of real-world events. In a crisis situation such as floodings, fires, storms or shootings, people nowadays report and discuss about their observations and opinions in Social Media [4] [3]. Some studies show that these amount of data help to detect the incidents [5] or to analyze the information reported by the people [1]. Tweets reflect useful event information for a variety of events. These event messages can provide a set of unique perspectives, regardless of the event type, and users sometimes report news prior to the traditional news media.

The first step of this process is the classification of tweets at high-level (by information type). It is usual to work with an specific ontology, called MOAC [1]. MOAC, the Management of a Crisis vocabulary, is a lightweight vocabulary aiming to provide terms to enable practitioners to relate different "things" in crisis management activities together as Linked Data.

We have applied different approaches, testing WordNet synonyms, spelling correction and Word embeddings [2].

Chapter 2 present the data definition and analysis. Chapter 3 describes our system and the approaches, and the last chapter shows the results obtained and the conclusions.

[1] available at http://observedchange.com/moac/ns/

2 DATA ANALYSIS

The data provided by the organization is based on a selection of incidents or events of different types. For each incident, a stream of tweets related to it has been collected using hashtags and keywords.

In 2018, TREC-IS is focused in the following 25 high-level classes:

- Request
  - Goods Services
  - Search And Rescue
  - Information Wanted
- Call To Action
  - Volunteer
  - Donations
  - Move People
- Report
  - First Party Observation
  - Third Party Observation
  - Weather
  - Emerging Threats
  - Significant Event Change
  - Multimedia Share
  - Service Available
  - Factoid
  - Official
  - CleanUp
  - Hashtags
- Other
  - Past News
  - Continuing News
  - Advice
  - Sentiment
  - Discussion
  - Irrelevant
  - Unknown
  - Known Already

We have analyzed the training data from a statistical point of view. Table 1 shows the training classes distribution.

As we can see the training collection is not well balanced, since there are classes with more than 120 tweets while others have only a few examples, insufficient to describe those classes or for a machine learning (ML) system to learn.

The goal of this task is for systems to categorize the tweets in each event/incident's stream into different information feeds that

Class	#	%
ContinuingNews	232	19,17%
Irrelevant	143	11,82%
Factoid	130	10,74%
Sentiment	124	10,25%
MultimediaShare	115	9,50%
KnownAlready	99	8,18%
Discussion	44	3,64%
Official	41	3,39%
Weather	38	3,14%
Advice	37	3,06%
SignificantEventChange	32	2,64%
EmergingThreats	32	2,64%
FirstPartyObservation	28	2,31%
MovePeople	24	1,98%
Unknown	20	1,65%
Donations	14	1,16%
ThirdPartyObservation	14	1,16%
ServiceAvailable	13	1,07%
PastNews	12	0,99%
InformationWanted	10	0,83%
Hashtags	4	0,33%
CleanUp	2	0,17%
Volunteer	2	0,17%
Total	1,210	100%

Table 1: TREC-IS training dataset: tweets distribution

might be consumed by different public safety personnel or used for post-event analysis.

3 SYSTEM OVERVIEW

Supervised learning algorithms demand for a valid application certain requirements that sometimes are difficult to meet. One of the most difficult to overcame in some cases is the need for a large and varied learning data set. When there is lack of data, two main strategies can be followed: *transfer learning* and *data augmentation*.

We have applied different strategies to increase data, using a traditional classification architecture, with the following components.

(1) **Getting the tweets**. Given the list of ids of the training and test tweets, the system used *twarc*[2] to download the data of each tweet id.
(2) **Preprocessing**. Each tweet was preprocessed as usual (filtering, repeated characters, punctuation marks, stopwords removal, stemming, etc). The preprocessing was made using TextBlob[3]. We tested spelling correction, based on Peter Norvig's module.
(3) **Topic expansion with synonyms**. Considering the low number of words in a tweet, we wanted to test the expansion of the context of the tweet using WordNet synonyms.
(4) **Topic expansion with word embeddings**. It is the collective name for a set of language modeling and feature learning techniques in natural language processing where words or phrases from the vocabulary are mapped to vectors of real numbers. It aims to quantify and categorize semantic similarities between linguistic items based on their distributional properties in large samples of language data. Based on the Wikipedia English files, we expanded each term of the preprocessed tweet with three related words. We used three new terms based on previous experiments.
(5) **Training**. The core framework functionality is triggered by an incident detection module based on a machine learning, SVM in our case.
(6) **Test**. When the test dataset was processed, we run it against each training model, obtaining the prediction class.

The experiments carried out have the following features:

(1) **Run1**: **baseline run**. Preprocessing without spelling correction, not topic expansion
(2) **Run2**: Preprocessing without spelling correction, topic expansion with **WordNet synonyms**
(3) **Run3**: Preprocessing with **spelling correction**, not topic expansion
(4) **Run4**: Preprocessing without spelling correction, topic expansion with **Word Embeddings**

After the first runs with the training dataset, and the analysis of the results our first decision was to delete some of the high-level classes. Specifically those that didn't have a number of tweets so that the automatic classifier could learn (*Discussion, FirstPartyObservation, PastNews, ThirdPartyObservation, Unknown, InformationWanted, ServiceAvailable, Volunteer, Hashtags*).

4 EVALUATION AND CONCLUSIONS

The metrics used for the evaluation are the usual ones: precision, recall, F1 and accuracy. Since a tweet can be categorized into one or more classes, to evaluate the quality of the runs the organization used a multi-type evaluation (categorization performance per information type in a 1 vs All manner) and a any-type evaluation (if the system assigned any of the categories that the human assessor selected for that tweet). The organization reported together with the evaluation of each run the median performance across participants.

Tables 2 and 3 show the overall performance and the overall performance (micro average) for each run.

Under the **multi-type** evaluation, the categorization performance is calculated per information type in a 1 vs All manner. A system is considered to have categorized a tweet for a category correctly if both the system and human assessor selected that category. The metrics used are:

•

Under the **any-type** evaluation, a system receives a full score for a tweet if it assigned any of the categories that the human assessor selected for that tweet. This is useful for providing a view on the overall performance of a TREC-IS system.The metrics used are:

•

Analyzing in a general way the results obtained we can conclude, as we had already foreseen in the analysis of the training data, that this is a complex task, which in many cases is complicated to solve

[2]available at https://github.com/DocNow/twarc
[3]available at https://textblob.readthedocs.io/en/dev/

Run	Precision	Recall	F1	Acc
1	0,1729	0,0726	0,0786	0,9029
2	0,1307	0,0696	0,0768	0,8987
3	0,1825	0,0712	0,0767	0,9023
4	0,1782	0,0753	0,0824	0,9025
MP	0,1827	0,0784	0,0824	0,0899

Table 2: Overall performance (multi-type)

Run	Precision	Recall	F1	Acc
1	0,5064	**0,5302**	**0,5181**	**0,3843**
2	0,4189	0,4979	0,4550	0,3317
3	0,5019	0,5129	0,5074	0,3762
4	**0,5297**	0,4849	0,5063	0,3785
MP	0,3977	0,6164	0,4774	0,3384

Table 3: Overall performance (micro average, any-type)

by humans, which means that an automatic system will not achieve good results.

Our system, by discarding those categories under-represented in the training data, has been affected in the recall value, but this has allowed the average values of the rest of participants to be reached and even increased.

In particular, the best precision value obtained with our system (0,5297) with the run4, significantly improves the median performance (0,3977). This is also the case for F1 and accuracy, although not with such a difference.

In a depth analysis, at topic level, Table 4 shows the results obtained for the best five topics.

Although the variation of results between the baseline case and the rest of the experiments is not significant, it can be verified that the topics that have achieved the best values are those that have been trained the most. We could place a threshold of 100 examples for a topic in training, as a measure for the classification system to work correctly.

Likewise, we can verify that in most of the analyzed cases, the best results have been obtained by applying spelling correction and word embeddings, and that the use of WordNet synonyms has introduced more noise, obtaining worse results.

As future work we will analyze the use of deep learning with a more adapted neural network system, introducing more training data for some of the topics and analyzing the behavior.

Topic	Run	Precision
CallToAction-Donations	1	0,74
CallToAction-Donations	2	0,22
CallToAction-Donations	3	0,79
CallToAction-Donations	4	0,77
Report-Weather	1	0,62
Report-Weather	2	0,65
Report-Weather	3	0,76
Report-Weather	4	0,63
Other-Sentiment	1	0,66
Other-Sentiment	2	0,66
Other-Sentiment	3	0,64
Other-Sentiment	4	0,66
Report-MultimediaShare	1	0,40
Report-MultimediaShare	2	0,37
Report-MultimediaShare	3	0,41
Report-MultimediaShare	4	0,51
Other-ContinuingNews	1	0,44
Other-ContinuingNews	2	0,36
Other-ContinuingNews	3	0,43
Other-ContinuingNews	4	0,49

Table 4: Precision values per run and topic (best five topics)

ACKNOWLEDGMENTS

This work has been partially supported by a grant from the Ministerio de Educación Cultura y Deporte (MECD - scholarship FPU014/00983), Fondo Europeo de Desarrollo Regional (FEDER) and REDES project (TIN2015-65136-C2-1-R) from the Spanish Government.

REFERENCES

[1] Devin Gaffney. 2010. Iranelection: quantifying online activism. In *In Proceedings of the Web Science Conference (WebSci10.*

[2] Yoav Goldberg and Omer Levy. 2014. Word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722. arXiv: 1402.3722. http://arxiv.org/abs/1402.3722.

[3] Akshay Java and Tim Finin. Why we twitter: understanding microblogging usage and communities. ().

[4] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? ().

[5] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *In Proceedings of the Nineteenth International WWW Conference (WWW2010). ACM.*