BJUT at TREC 2018: Incident Streams Track

Ning Lu, Hesong Wang, Zhen Yang*
College of Computer Science, Beijing University of Technology, China
yangzhen@bjut.edu.cn

Abstract

In this paper we will introduce our work on the 2018 TREC real-time event flow test task. With the development of social media, more and more people choose to use social media to share their lives. Similarly, when encountering unexpected situations such as fires, earthquakes, flash floods, tsunamis, mudslides and other natural disasters or shootings, robberies and other emergencies, people like to release the progress of the disaster situation or event through social media. This task is to filter the information of such natural disasters or emergencies through text detection, and to classify the information, and finally to report the marked information to relevant staff according to different priorities. Let the staff know about the progress of the incident and the local real-time situation in case of rescue. This article will introduce the framework and methods of the classification system, as well as the experimental results.

Introduction

In the Internet age, people like to use the Internet to record and share their lives. Such as WeChat, Weibo, Twitter, Instagram, Facebook, etc. Twitter is one of the most popular social networking platforms, with more than 500 million users, and millions of tweets posted online every day. In addition to sharing information about daily life, the tweets also contain tweets for emergencies such as fires, earthquakes, flash floods, tsunamis, mudslides, typhoons and other natural disasters or shootings, robberies and other emergencies. The information was sorted out in the first place and it was very helpful for the relevant staff to carry out rescue work.

Based on this, TREC 2018 Incident Streams Track task is to quickly filter out the information of emergencies from massive tweets, according to different events such as: torrents, tsunami, typhoon and so on. After that, the tweet is classified twice: if the tweet is about the disaster situation (including time, place, disaster area, etc.), after the disaster (including location information, item demand, volunteer demand, etc.), the disaster occurs. Early warnings after the occurrence of disasters (such as typhoon warning, earthquake warning, aftershock warning) and so on. After the secondary classification is completed, the tweet needs to be classified for the third time: each tweet is classified into four levels: severe, high, medium, and low. Finally, the tweet is scored according to the corresponding score calculation rule, and the result is output. Since the official classification of the mission has already completed the first classification, this article only describes the second classification and the third classification.

The paper is organized as follows: the second section introduces the classification method of this task, the third section shows the experimental results, and the fourth section summarizes.

Incident Streams System Framework

This chapter will focus on the classification system design used to complete the task. The system consists of three parts: query expansion module, training model module and prediction module design. Figure 1 shows our system framework.

- Query extension module

  Since the Incident Streams Track mission is the first year of this year, the number of training sets given by the official is very small. According to the official example of the ontology label and the training set, we have expanded the query. According to the keywords in the sample, we use the keyword search form to crawl the content on Twitter, BBC News, Fox News, and expand the training set corpus. Among them, BBC News and Fox News are designed to cover more vocabulary and solve the problem of less short-covering vocabulary. The parent of each keyword is the official ontology tag, so all data is tagged with the corresponding ontology tag as a training corpus.

- Training model module

  First, the corpus is preprocessed. The content crawled by Twitter is real-time. There will be a large number of forwarded tweets. The content is basically the same. Therefore, only the same corpus content will be kept in the preprocessing, and the rest of the tweets will be discarded and then we remove stop words from the corpus and links to prevent interference during training.

  After that, word frequency statistics are expected to be converted into word frequency matrices. Since there are some words with higher frequencies in the tweet, they are not meaningful in the actual features. So after the word frequency matrix is established, we do a TD-IDF transformation on the matrix. Finally, the matrix is input into the SVM model to train and save the model.

Table 1: Ontology Quantitative Score

label	value
Request-GoodsServices	5
Request-SearchAndRescue	5
Request-InformationWanted	4.5
CallToAction-Volunteer	4
CallToAction-FundRaising	4
CallToAction-Donations	4
CallToAction-MovePeople	4
Report-FirstPartyObservation	3
Report-ThirdPartyObservation	3
Report-Weather	3.5
Report-EmergingThreats	4
Report-SignificantEventChange	3.5
Report-MultimediaShare	2
Report-ServiceAvailable	3.5
Report-Factoid	3
Report-Official	3
Report-CleanUp	3
Report-Hashtags	2
Other-PastNews	1
Other-ContinuingNews	2
Other-Advice	2
Other-Sentiment	1
Other-Discussion	1
Other-Irrelevant	0.5
Other-Unknown	1

Table 2: Graded Quantified Score

level	value
Critical	5
High	4
Medium	3
Low	2

When used, two models are output, corresponding to the model classified by ontology and the model of hierarchical classification.

- Prediction module

  We randomly selected the expected 1/4 corpus as the test set and 3/4 as the training set in the training expectation. The working principle of this module is shown in Figure 1. The corpus is entered into the model above, and the two models above put the corpus on the label of the ontology and the label of the grading.

  Since the official request requires a quantitative ranking of the final output, it is considered that the ontology label and the grade label are scored, as shown in Table 1 and Table 2. The score corresponding to each tweet is calculated as: Tweet score = grade label score * ontology tab score.

Table 3: Results

	myrun1	myrun2
Precision	0.18	0.20
Recall	0.88	0.59
F1	0.30	0.30
Accuracy	0.17	0.19

Submitted Runs and Experiment Results

We submitted two predictions, the first being the results of all model classification predictions (myrun1). The second is the result of some human intervention (myrun2) on the tweets with low prediction accuracy. The results are shown in Table 3.

Conclusion

According to the experimental results, we can see that the experimental results are not ideal. By looking for the reasons, we think it is a problem of data sources. Due to tweets or news crawled by keywords, it is not able to match the tags very well, resulting in some corpora being not the best corpus of the tag, and even some corpora cannot reflect the tag, causing interference.

Corpus crawling through keywords does not cover all situations well. Because the short text expression is relatively flexible, an expression can express the mood of the author of the tweet, such as the fact that our corpus coverage is not comprehensive enough. The result is biased.

Models mixed by model prediction and human intervention can improve the prediction effect, but the cost of human intervention is relatively high and is not suitable for promotion.

References

Wang, K., and Yang, Z. 2016. Bjut at trec 2016: Real-time summarization track. In TREC.

Tan, H.; Luo, D.; and Li, W. 2016. Polyu at trec 2016 realtime summarization. In TREC.

Lin, J.; Roegiest, A.; Tan, L.; McCreadie, R.; Voorhees, E.;and Diaz, F. 2016. Overview of the trec 2016 real-time summarization track. InProceedings of the 25th Text REtrieval Conference, TREC, volume 16.

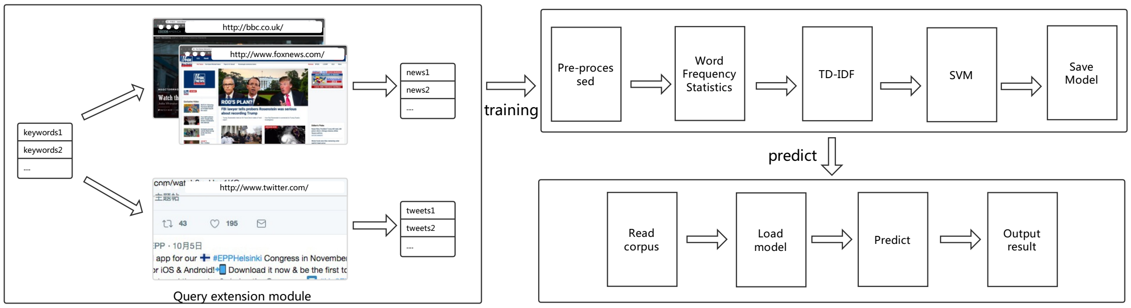

Figure 1: System Framework.