Automatic Retrieval of Actionable Information from Disaster-related Microblogs

Saloni Baweja
Indraprastha Institute of Information Technology
Delhi, India
saloni16049@iiitd.ac.in

Aniya Aggarwal
IBM Research
India
aniyaagg@in.ibm.com

Vikram Goyal
Indraprastha Institute of Information Technology
Delhi, India
vikram@iiitd.ac.in

Sameep Mehta
IBM Research
India
sameepmehta@in.ibm.com

ABSTRACT

This paper discusses our work submitted to FIRE 2017 IR-MiDis Track [3]. The goal was to extract actionable information from the micro-blogs i.e. tweets which can be leveraged to provide aid and help during disaster events. The two tasks addressed in this work are, first, extraction of useful information such as the need or availability of various resources and second, finding tweets that express the need and availability of the same resources. Our approach is based on leveraging a mix of linguistics and machine learning techniques. The evaluation scores of the submitted runs are reported in terms of Precision@100, Recall@100 and MAP. The average MAP score is reported to be 0.1304 for the identification of need and availability tweets. The score for the matching task is reported in terms of the F-score which came out to be 0.2424.

General Terms

Information Retrieval, Need-Tweet, Availability-Tweet

Keywords

Disaster Management, Resource Need, Resource Availability, Information Retrieval, Machine Learning, Microblogs, Nepal, Earthquake, POS Tag, Fully Automated

1. INTRODUCTION

In this digital era, the increasing use and popularity of various social media platforms has enabled people to connect worldwide in a fast and efficient manner. Microblogging applications like Twitter play a significant role in disseminating real-time updates among the masses. Especially during the time of emergencies or natural calamities, such microblogging sites are well leveraged by the NGOs, agencies, relief providers and the general public for the exchange of information. The real-time updates posted during the disaster events, if exploited properly can help aid the victims and guide the agencies to perform relief operations in an efficient and effective manner.

.

2. TASKS

This work primarily discusses our automated approach to extract actionable information from a set of available tweets posted during the Nepal earthquake in year 2015 for post-disaster relief process. We explore the use of several machine learning algorithms and linguistics to design such an automated retrieval system in order to address the following two tasks.

1. Identify the tweets indicating the need and availability of various resources like food, water, electricity, medical aid, shelter, mobile or Internet connectivity etc.
2. Match the set of need tweets with appropriate availability tweets.

Any tweet that specifies the need or requirement of any of the aforementioned resources is termed as a *need-tweet*. This category also encompasses the tweets which do not directly specify the need, but point to scarcity or non-availability of some resources. Whereas the tweet that expresses the accessibility or availability of the resources is tagged as an *availability-tweet*. This class not only includes the tweets informing about the actual availability of the resources but also includes the ones which inform about potential availability in future, such as resources being transported or dispatched to the disaster-struck area. Below are the samples of a need-tweet and an availability-tweet identified from the provided dataset.

1. *Plz provide medicine,blood,food,clean water,shelter and moral support to people of #Nepal #NepalEarthquake*
2. *UP govt to send relief material in 21 trucks to quake-hit Nepal,comprising 10 trucks mineral water,10 trucks biscuits and 1 truck medicines*

In this case, Tweet 1 is a need-tweet while tweet 2 is an availibility-tweet. Also, since Tweet 1 specifies the need of water, food and medicines while Tweet 2 specifies the availability of all of these, therefore they both correctly match each other as well.

3. DATASET

The entire dataset provided during the track contains 70k tweets posted during the course of Nepal Earthquake, April

2015. The tweets in the provided collection are written in a mix of three languages, namely, English, Hindi, and Nepali. The entire data was made available in two phases.

1. In the first phase, a *training dataset* comprising of approximately 20k tweets labelled as either *need*, *availability*, and *others* was provided.

2. In the second phase, a *test set* containing around 50k unlabelled tweets was made available.

We have not used any other data resources apart from the above mentioned ones for classification.

Majority of the tweets in the provided training collection of 20k tweets had *others* tag assigned. Therefore, only a small fraction of the training set had *need* or *availability* tag assigned. Such tweets were less than 1000 in number, each for the need and availability category. This posed a major challenge in building the classification model due to availability of less labelled data. Furthermore, the skewed nature of labelled data was an another challenge as the count of the availability-tweets was a lot more than that of the need-tweets which could potentially bias the classification model.

4. METHODOLOGY

This section discusses the overall design and implementation of our approach in detail. We leverage the capabilities of machine learning algorithms and linguistics to implement the two aforementioned tasks which are well discussed in the subsequent subsections.

4.1 Task 1 : Tweets Classification

This task concerns with identifying the tweets that specify the need, non-availability or scarcity of various resources like food, water, electricity, medical aid, shelter, mobile or Internet connectivity etc.

The overall process can be divided into three non-overlapping phases, namely *Preprocessing*, *Feature Selection*, and *Model Selection*.

4.1.1 Preprocessing

This phase involves performing all the clean-up jobs on the provided tweets labelled as either *need* or *availability* in the training set. All the words starting with hashtags along the usernames starting with @ are first pruned from every tweet. The hashtags such as *#Nepal, #earthquake*, etc. are removed since they appear in majority of the tweets belonging to both of the need and availability categories and therefore, are not of much help for training the classification model. All the URLs i.e. words starting with HTTP or http present in the tweets are also removed. We next identify the duplicate tweets and the retweets available in the set by gauging their cosine similarity and exclude them too. The duplicates removal further reduced the count of our need and availability tagged tweets to be fed in the classifier.

4.1.2 Feature Selection

After cleaning up the dataset, the next step is to extract features from the labelled tweets to train the binary classification model. While manually inspecting the tweets, we realized that the POS tags or the tense of the words can help a lot to infer if the tweet relates to the need or availibility category. For instance, let us consider the following two sample tweets from the dataset.

1. *@joanna_udo: Contact Youth for Blood if in need of blood in Nepal #NepalEarthquake #NepalQuake #PrayForNepal*

2. *@BDUTT Need more resources and personnel from Army for restoring Power / Bridges / Roads and to have substantial presence in Nepal*

Here, Tweet 1 falls in the category of availability tweets while Tweet 2 belongs to the need category. The word *need* appears in both the tweets but with a different context and POS tag. The POS tag of the word *need* in Tweet 1 is *NN* which refers to a noun whereas in Tweet 2, its *VBD* which indicates that it is a verb. The same lemma with different POS tags in different tweets tends to have different need/availability tag. Therefore, our features were constructed using both the word and its POS tag.

We use Stanford CoreNLP library [1] to identify the POS tags along with its lemma of all the words in the tweet set. The words in the tweets are considered in the form of *wordlemma_POSTag*. Each tweet is represented as a dictionary of key-value pairs where the key corresponds to *wordlemma_posTag* and the value corresponds to 1 or 0 indicating its presence or absence in the tweet. It is to be noted that we do not consider the stop words along with its POS tag while creating this dictionary for every tweet. This collection or list of dictionaries for the entire labelled tweet set is transformed using the *Dict Vectorizer*. It is used to convert the features into an array (required shape) to feed to the learning models. For instance, the following tweet
@TheEllenShow people are running out of water n foods. Please help #nepal. #HELPNEPAL #NepalEarthquake
is transformed to
food_NNS: 1, help_VB: 1, water_NN: 1, run_VBG: 1, please_VB: 1, people_NNS: 1
using this vectorizer.

4.1.3 Model Selection

After extracting the features, it is time to feed them in a binary classification engine that after learning assigns a need or *availability* class to every tweet in the test set. After experimenting with several learning algorithms, we finally picked *Logistic Regression* [2] to be used for this classification task. The model is trained using the set of features extracted from non-duplicate need and availability tweets available in the training set. We further reassigned the identified classes of the tweets based on applying a threshold on the identified class probabilities which was decided by experimentation. In our case, it was set as 0.4. All the tweets whose class probabilities were unable to cross the threshold of 0.4 are categorized as *others*. This was done to handle the cases where a tweet may not belong to any of the need or availability classes.

4.2 Task 2: Matching Need and Availability Tweets

The goal of this task is to find at most five availability tweets against an identified need tweet. Apart from the preprocessing done for the classification task, stop words are also removed from the identified need and availability tweets. The need and availability tweets describe the requirement or accessibility of resources like food, water, and electricity. These words or resources, in most of the cases, have a POS tag of Nouns. We therefore transform every identified need

Table 1: Results for Task 1

Category	Precision@100	Recall@100	Map
Availability-Tweets	0.6700	0.1929	0.1233
Need-Tweets	0.4100	0.3864	0.1376

Table 2: Overall Results for Task 1

Run Type	Average MAP
Automatic	0.1304

and availability tweet into a list of the Noun words that occur in the tweet. However, some of the Proper Nouns like Nepal, Delhi, Kathmandu, and India occur in both need and availability tweets. Therefore, we identified such frequently occurring proper noun words and eliminated them from the tweets to facilitate the matching process. For every transformed need tweet, we find its cosine similarity against every transformed availability tweet. At most top five availability tweets having maximum cosine similarity score with a need tweet, which cross a certain similarity threshold are identified for that need tweet. The similarity threshold is set as 0.7 in our approach as inferred on the basis of experimentation. This brute force searching is employed in our run submission 1.

In our run submission 2, we follow a greedy approach and don't search or process all the availability tweets for a need tweet. The search stops as soon as it finds the first five or lesser availability tweets with a cosine similarity score greater than our set threshold of 0.7.

5. EVALUATION

The gold-standard is generated using *manual runs*. As mentioned in the IRMiDis Track, the human assessors are given the same set of tweets, indexed in a search engine and are asked to identify the need and availability tweets for Task 1 evaluation. For Task 2, the assessors are asked to identify matching availability-tweets against each need-tweet. To identify relevant tweets or matches which the annotators may not have found, polling is used over the participants' runs. The run submissions are evaluated against these gold standards. Measures like precision, recall, Mean Average Precision (MAP), and F-score are used for the evaluation of runs.

5.1 Task 1 : Tweets Classification

The results for Task 1 are evaluated using metrics like *Precision*, *Recall*, and *MAP*. As explained in the IRMiDis Track, *precision* is defined as the fraction of actual need or availability tweets retrieved, while *recall* denotes the fraction of all need or availability tweets (out of all the tweets in the gold standard) that could be retrieved by a certain methodology. MAP is an another metric used which is Mean Average Precision (MAP) considering the retrieved ranked list.

The results of our submitted automated run for Task 1 are shown in Table 1 both for the need and availability tweets. Our automatic run submission enjoyed 5th rank among the other submissions with an average MAP score of 0.1304 as reported in Table 2 as well.

Table 3: Results for Task 2

Run Submission	Precision@5	Recall	F-Score	Rank
Run 1	0.1758	0.3677	0.2379	3
Run 2	0.2081	0.2904	0.2424	2

5.2 Task 2: Matching Need and Availability Tweets

The results of the matching task were evaluated using metrics like *Precision@5*, *Recall*, and *F-Score*. As mentioned in the IRMiDis Track, *Precision@5* means that for every need-tweet that is correctly identified, it is checked that how many of the five matches reported are correctly matched. *Recall* on the other hand indicates the fraction of overall need-tweets which can be correctly matched by at least one availability-tweet.

The results of our submitted automated runs are shown in Table 3. Our automatic run submission 2 and 1 were placed at position 2 and 3 respectively among the other submissions with F-Score of 0.2424 and 0.2379 respectively.

6. CONCLUSION

In this work submitted to IRMiDis Track, we used a mix of linguistics and machine learning models to automatically identify the tweets indicating need and availability of resources in a disaster affected area. The features used to train our classifier considered the word lemma along with its POS tag in every labelled tweet. We also discuss an approach to automatically uncover the correspondence between the identified need and availability tweets. For a tweet indicating the need of a particular resource, we find the relevant tweets indicating its availability by computing its cosine similarity score. Every tweet in such a case is translated in to a bag of noun words present in it. As a future extension of this work, we plan to explore more sophisticated approaches to build features to train the classifier. The relative sequence of words in the tweets for example, may play a significant role in improving the performance of the model which may be incorpoarted in future.

References

[1] URL: `https://nlp.stanford.edu/software/tagger.shtml`.

[2] URL: `http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html`.

[3] Moumita Basu et al. "Overview of the FIRE 2017 track: Information Retrieval from Microblogs during Disasters (IRMiDis)". In: *Working notes of FIRE 2017 - Forum for Information Retrieval Evaluation*. CEUR Workshop Proceedings. Bangalore, India: CEUR-WS.org, Dec. 2017.