An Approach for Event Detection from News in Indian Languages using Linear SVC

Fazlourrahman Balouchzahi, H L Shashirekha

Department of Computer Science, Mangalore University, Mangalore - 574199, India

Abstract

Automatically handling the enormous amount of text data that is being generated with mind-blowing speed is an ongoing work in text processing for various applications. Event Detection (ED) is one such application that aims to extract information about events in a given text based on the words which indicate the events. It acts as a preprocessing step for various Natural Language Processing (NLP) applications such as relation extraction, topic modeling, and decision making. In this paper, we, team MUCS, present an approach using Linear SVC to identify pieces of text indicating events and then classifying those events into predefined categories using n-grams, suffix and prefix features. The model has been submitted to Event Detection from News in Indian Languages (EDNIL) task in Forum for Information Retrieval Evaluation(FIRE 2020).

Keywords

Event Detection, Linear SVC, NLP, N-grams

1. Introduction

Something that happens at a specific time and place is described as an event. It could be a natural event such as earthquake, flood or a manmade event such as accident, killing. For example, the news article, “7 people have died in coastal Kerala and 6 in Tamil Nadu and nearly 90 fishermen are missing as a depression above the Bay of Bengal turned into a cyclonic storm Ockhi” describes a natural event ‘cyclone’. Monitoring the events over time is helpful for organizations to analyze the situation and take the necessary action. Information about such events not only appears in newspapers and new channels but also in the online version of these newspapers and new channels which will be updated regularly. Due to rapid growth of the news articles that are being published daily, it becomes truly impossible to manually extract the events and understand them. Further, extracting relevant news about the events manually is not only time consuming but cumbersome and error prone also. This demands algorithms for Event Detection (ED) that automatically detects the events from the given news data which is basically an unstructured text [1]. ED is an Information Extraction task that acts as a preprocessing step for many NLP applications such as relation extraction, topic modeling and so on [2].

Various studies related to ED have been taken up by researchers. However, most of the research work is focused on resource rich languages such as English ignoring the Indian lan-

Forum for Information Retrieval Evaluation, December 16–20, 2020, Hyderabad

frs_b@yahoo.com (F. Balouchzahi); hlsrekha@gmail.com (H.L. Shashirekha)
https://mangaloreuniversity.ac.in/dr-h-l-shashirekha (H.L. Shashirekha)
0000-0003-1937-3475 (F. Balouchzahi)

guages which are resource poor. To promote NLP in Indian languages FIRE 2020 has called for 'Event Detection from News in Indian Languages (EDNIL) as a Shared Task that include two tasks namely, Event Detection and Event Frame extraction. While the aim of ED is to identify a piece of text from news articles that contain a disaster event and classifying it into one of two classes, Manmade Disaster and Natural Disaster, Event Frame extraction aims at building an Event Frame that includes:

1. Extracting words associated with the type of event from the given text and
2. Extracting sub-type of events based on the type of event. The event Manmade disaster has sub-types as CRIME, RIOTS, AVIATION_HAZARD, ACCIDENTS, SUICIDE_ATTACK, FIRE etc. and Natural Disaster has sub-types such as FOREST_FIRE, HURRICANE, COLD_WAVE, TORNADO, STORM, HAIL_STORMS, BLIZZARD, AVALANCHES, etc.
3. Casualties: Number of people injured or killed/damages to the properties
4. Time: When did the event happen
5. Place: Where did the event happen
6. Reason: Why and how the event happened

More details about the tasks are given in the shared task website[1] and reference paper [3]. In this paper, we, team MUCS, present an approach using Linear SVC to identify pieces of text indicating events and then classifying those events as Manmade Disaster or Natural Disaster using n-grams, suffix and prefix features.

2. Literature Review

Several studies have been carried out by various researchers in ED, some of the relevant works are mentioned below: Jianshu et al. [4] presents a study on ED by clustering wavelet-based (EDCoW) signals. They build signals for individual words by applying wavelet analysis that provides precise measurements regarding when and how the frequency of the signal changes over time on frequency-based raw signals of the words and then filters away the trivial words by looking at their corresponding signal auto-correlations. The remaining words are clustered to form events with a modularity-based graph partitioning technique. On a dataset collected from Twitter containing 43, 31, 937 tweets and 6, 38, 457 unique words, they obtained precision in range of 14.30% to 76.20%. An ED system on multi-lingual social streams proposed by Yaopeng et al. [5] automatically detect events and generate evolution graph in multilingual hybrid-length text streams including English, Chinese, French, German, Russian and Japanese. The authors used 8-tuple to describe an event for correlation analysis and evolution graph generation and obtained an f1 score of 0.7332 on a raw dataset including Twitter, Weibo, WeChat, Worldwide Publishing House and forum stored in HBase and indexed by Elasticsearch. Pankaj et al. [6] presented a system called MIC-CIS in the fine-grained Propaganda Detection Shared Task 2019[2]. The shared task includes two tasks namely, sentence (SLC) and fragment level (FLC) propaganda detection. The authors have explored neural architectures namely, CNN, LSTM-CRF and BERT and also used different linguistic features such as part-of-speech, named

[1] https://ednilfire.github.io/ednil/2020/index.html

[2] https://propaganda.qcri.org/nlp4if-shared-task/

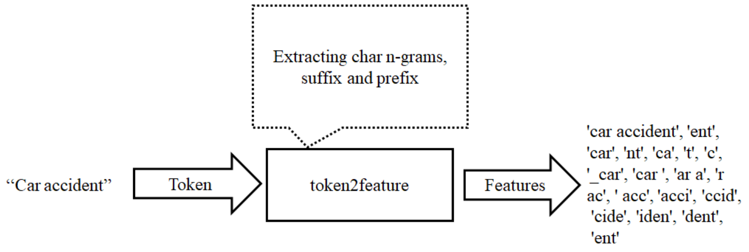

Extracting char n-grams, suffix and prefix
"Car accident"
Token
token2feature
Features
'car accident', 'ent',
'car', 'nt', 'ca', 't', 'c',
'_car', 'car ', 'ar a', 'r
ac', ' acc', 'acci', 'ccid',
'cide', 'iden', 'dent',
'ent'

Figure 1: Feature generation module

entity, readability, sentiment, emotion, etc. They have also designed multi-granularity and multi-tasking neural architectures for both the sentence and fragment level propaganda detection. Additionally, different ensemble schemes such as majority-voting, relax-voting, etc. have been investigated to boost overall system performance. The proposed model obtained 3rd and 4th rank in the shared task with f1 score of 0.1999 and 0.6231 for FLC and SLC tasks respectively. TwitterNews a real time ED system presented by Mahmud et al. [7] combines random indexing based term vector model with locality sensitive hashing, which aids in performing incremental clustering of tweets related to various events within a fixed time. The proposed system consists of Search and EventCluster modules. Search module allows fast retrieval of the neighboring tweets of the input tweet for text similarity comparison by using the adapted variant of the Locality Sensitive Hashing (LSH) approach [8] and Random Indexing [9]. EventCluster module incrementally clusters the tweets discussing the same topic and produces a set of candidate events. TwitterNews obtained a recall of 0.87 and precision of 0.72 on Events2012 corpus containing 120 million tweets. Along with this corpus, 506 events and the relevant tweets for these events are provided as ground truth.

3. Methodology

The proposed approach accepts the train data in the form of XML files consisting of news articles as input and is cleaned by removing unnecessary characters such as ,_, =, @, ,% and stopwords. The remaining data is split into tokens using XML tags such that a token represents a single word or group of words that indicates a disaster. Then a feature generation module is used to generate char n-grams (n = 1, 2, 3, 4), suffix and prefix of length k (k=1, 2, 3) features for tokens. These features are transformed to vectors by CountVectorizer which are in turn used to train a Linear SVC classifier to detect events. Figure 1 represents the workflow of the feature generation module. The ED module accepts the test data in the form of XML file as input, cleans the data by removing unnecessary characters and stopwords and generates the feature vectors. These feature vectors are given as input to the Linear SVC classifier built using the train set which will detect the events and generates an XML file as per the requirement of the organizers consisting of the pairs (event, label) where the event is any of the subtype such as CRIME, RIOTS, AVIATION_HAZARD, ACCIDENTS, SUICIDE_ATTACK, FOREST_FIRE, HURRICANE and label of an event is either MAN MADE DISASTER or NATURAL DISASTER. The ED module is shown in Figure 2.

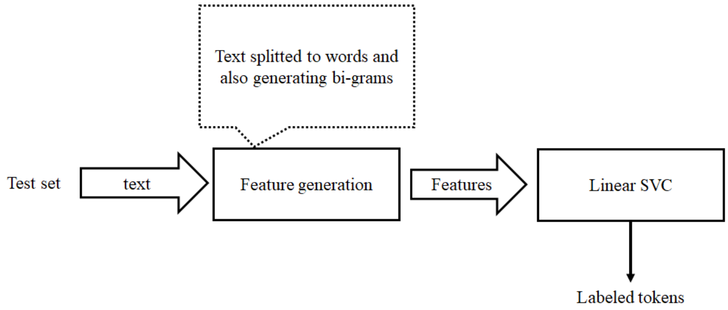
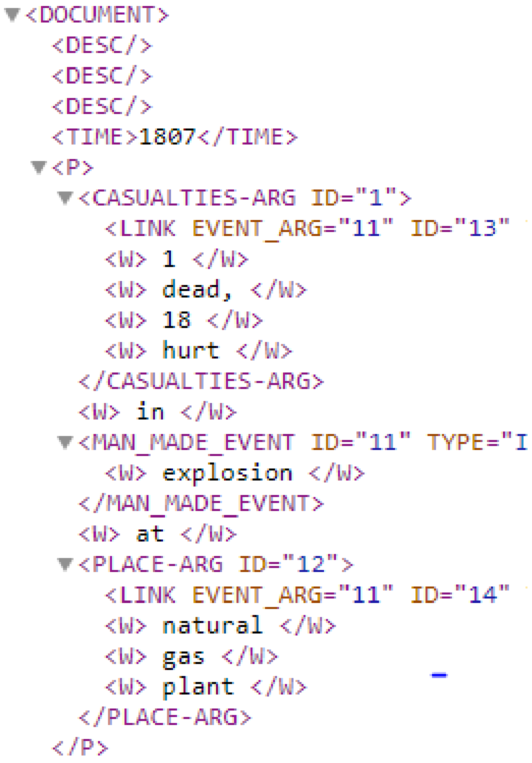


Figure 2: Event detection module



```
▼<DOCUMENT>
   <DESC/>
   <DESC/>
   <DESC/>
   <TIME>1807</TIME>
 ▼<P>
   ▼<CASUALTIES-ARG ID="1">
       <LINK EVENT_ARG="11" ID="13"
       <W> 1 </W>
       <W> dead, </W>
       <W> 18 </W>
       <W> hurt </W>
     </CASUALTIES-ARG>
     <W> in </W>
   ▼<MAN_MADE_EVENT ID="11" TYPE="I
       <W> explosion </W>
     </MAN_MADE_EVENT>
     <W> at </W>
   ▼<PLACE-ARG ID="12">
       <LINK EVENT_ARG="11" ID="14"
       <W> natural </W>
       <W> gas </W>
       <W> plant </W>
     </PLACE-ARG>
   </P>
```

Figure 3: Train set format

4. Experimental Results

The datasets provided by the EDNIL organizers for the shared task consists of news articles in English and four Indian languages namely, Hindi, Bengali, Tamil, and Marathi. Datasets for each language include train documents as XML files with specific tags and test documents as XML files with text body only. Description of datasets is explained in task website and a sample XML file used as train document is shown in figure 3. Distribution of data in the dataset is given in Table 1.

Table 1
Data distribution

Language		English	Hindi	Bengali	Marathi	Tamil
No. Documents	Train set	828	828	800	1035	1013
	Test set	206	194	204	265	257

Table 2
F1 score of all the participating teams

Team	Language				
	English	Bengali	Hindi	Tamil	Marathi
3 idiots	0.74	0.61	0.62	0.68	0.50
BUDDI_SAP	0.62				
ComMA	0.58	0.38	0.51		
MUCS	0.34	0.21	0.25	0.17	0.19
NLP@ISI	0.32	0.10			

EDNIL 2020 has two shared tasks for all the five languages. But, we have participated in only the first task of identifying a piece of text that indicates a disaster event and then classifying it as either MAN MADE DISASTER or NATURAL DISASTER for all the five languages. f1 scores of all the five teams who participated in this shared task is shown in Table 2. The results clearly show that only 2 teams participated for all the languages and team MUCS is one among them. No doubt that the performances of our models are less compared to others' models but, we have initiated to develop ED models for Indian languages which are very much required in the present context. These models can be improved by extracting features relevant to the events and also by experimenting on different classifiers.

5. Conclusion and Future work

EDNIL in FIRE 2020 is a shared task to detect events from news in Indian languages. We, team MUCS, submitted a base model using Linear SVC based on char n-grams, suffix and prefix features of tokens for all the five languages of Task 1 and our team was one of two teams who submitted results for all languages in task 1. Even though the performances of our models are less compared to others' models, we have initiated to develop ED models for Indian languages. Conducting experiments on ED task using different learning approaches such as Deep Learning and Transfer Learning will be the future work. The number of participants in EDNIL task in FIRE 2020 illustrates that it is not an easy task to identify events from Indian languages.

References

[1] Z. Saeed, R. A. Abbasi, O. Maqbool, A. Sadaf, I. Razzak, A. Daud, N. R. Aljohani, G. Xu, What's happening around the world? a survey and framework on event detection techniques on twitter, Journal of Grid Computing 17 (2019) 279–312.

[2] L. Hu, B. Zhang, L. Hou, J. Li, Adaptive online event detection in news streams, Knowledge-Based Systems 138 (2017) 105–112.

[3] B. Dave, S. Gangopadhyay, P. Majumder, P. Bhattacharya, S. Sarkar, S. L. Devi, Overview of the FIRE 2020 EDNIL track: Event Detection from News in Indian Languages, in: P. Mehta, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2020 - Forum for Information Retrieval Evaluation, Hyderabad, India, December 16-20, 2020, CEUR Workshop Proceedings, CEUR-WS.org, 2020.

[4] J. Weng, B.-S. Lee, Event detection in twitter., Icwsm 11 (2011) 401–408.

[5] Y. Liu, H. Peng, J. Li, Y. Song, X. Li, Event detection and evolution in multi-lingual social streams, Frontiers of Computer Science 14 (2020) 1–15.

[6] P. Gupta, K. Saxena, U. Yaseen, T. Runkler, H. Schütze, Neural architectures for fine-grained propaganda detection in news, arXiv preprint arXiv:1909.06162 (2019).

[7] M. Hasan, M. A. Orgun, R. Schwitter, Twitternews: real time event detection from the twitter data stream, PeerJ PrePrints 4 (2016) e2297v1.

[8] S. Petrović, M. Osborne, V. Lavrenko, Streaming first story detection with application to twitter, in: Human language technologies: The 2010 annual conference of the north american chapter of the association for computational linguistics, 2010, pp. 181–189.

[9] M. Sahlgren, An introduction to random indexing, in: Methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering, 2005.