Exploiting Transfer Learning and Hand-Crafted Features in a Unified Neural Model for Identifying Actionable Informative Tweets

ABU NOWSHED CHY\textsuperscript{1,a)} UMME AYMUN SIDDIQUA\textsuperscript{2,b)} MASAKI AONO\textsuperscript{3,c)}

Received: April 8, 2020, Accepted: October 6, 2020

Abstract: During emergencies and disaster situations, microblogging sites, especially twitter, can be used as a source of providing situational information needs. Monitoring and identifying informative tweets from tweet streams provide enormous opportunities for public safety personnel in coordinating aid operations as well as conducting the post-incident analysis. However, the brevity of tweets and noisy tweet contents makes it challenging to extract the situational information effectively and identify the tweets based on different information types. In this paper, we propose a neural network model with a naive rule-based classifier for actionable informative tweets classification. In our proposed neural architecture, we exploit the transfer learning features from a pre-trained sentence embeddings model along with a rich set of hand-crafted features to train a multilayer perceptron (MLP) network. In addition, we employ the state-of-the-art LSTM variants, nested LSTMs (NLSTMs) to capture the long-term dependency effectively. On top of nested LSTMs, we perform the convolution using multiple kernels (CMK) to obtain the higher-level representation of tweets. Experiments on the 2018 TREC incident streams (TREC-IS) dataset show that our proposed neural model learns the contextual information effectively and achieves the overall best result compared to the state-of-the-art methods.

Keywords: microblog incident streams, actionable information, crisis informatics, disasters, nested LSTMs, convolution using multiple kernels, transfer learning features, hand-crafted features.

1. Introduction

Microblog platforms such as twitter, sina weibo, etc. are rapidly moving towards a platform for informal user-generated information production and consumption. Among the several microblog services, twitter has become the most popular. The real-time nature of twitter plays an important role during a disaster period, such as earthquakes, wildfires, and so on. This is because the user-generated twitter posts during such events might be useful to serve the situational information needs \cite{1}. For example, we can consider a recent fire incident that happened in the world’s largest tropical rainforest Amazon. When the fire began in Amazon, we saw that lots of people posted a vast number of tweets about this issue on twitter. Upon observing such tweets, we saw that the actionable information shared by the people in twitter including the current situations of the incident, requesting a search and rescue operation, asking people to leave an area, asking for funding and donations, asking for volunteers, asking for service or physical goods, and reporting weather information. That is why safety personnel from government and non-government organizations are increasingly interested in identifying or categorizing such actionable informative tweets produced on twitter during a crisis period. But retrieving and identifying actionable information from twitter is regarded as a challenging information retrieval (IR) problem due to the specific nature of tweets.

To address the general real-time information retrieval (IR) problems in twitter, the text retrieval conference (TREC) introduced the microblog ad-hoc search task in 2011 \cite{2}. In contrast, TREC-2018 introduced an incident streams (TREC-IS) task designed specifically to tackle the challenges of utilizing information shared in microblog during an emergency period.

2018 TREC Incident Streams (TREC-IS) Task: The 2018 TREC-IS track focused on analyzing the social media posts, especially tweets and categorizing them into several incident-related classes. The curated posts are beneficial to the safety personnel to take the necessary actions or post-incident analysis. The main task of the track is formally defined as follows \cite{3}, \cite{4}:

Task: Classifying tweets by information type (High-level)

Given a tweet related to an incident event, a system needs to assign a high-level information type (i.e., one category per tweet). In the 2018 track, organizers only focused on the classification task and the primary evaluation measure was the classification F1 score, micro averaged over the different information types. The high-level information-types are defined in the 2018 TREC-IS incident ontology \cite{3}, \cite{4} and the incident events that are considered in 2018 track including earthquakes, typhoon/tornado, flood,
shooting, bombing, and bushfire/wildfire. We provide the detail description of the dataset, evaluation criteria, and performance analysis regarding this task in the Section 4.

Participants proposed several methods in the 2018 TREC-IS competition. Some participants addressed the problem using traditional deep learning based approaches (e.g., CNN, LSTM) [5], [6], whereas other participants utilized the classical supervised classifiers (e.g., SVM, Naive Bayes) in their method [5], [7], [8]. However, the lack of utilizing state-of-the-art deep learning techniques effectively hampered the performance of most of the participants’ methods.

The main contribution of this paper is that we propose a unified neural network model that incorporates the multilayer perceptron (MLP), multiple kernels based convolution, and nested LSTMs [9] model, where MLP model imputed with the transfer learning features and hand-crafted features. Our neural model learns the contextual information effectively and we show the efficacy of our method based on the experiments on 2018 TREC incident streams (TREC-IS) dataset.

The rest of the paper is organized as follows: Section 2 provides a detailed overview of prior research, which instigates us to contribute in this domain. In Section 3, we introduce our proposed framework. Section 4 includes experiments and evaluation as well as comparative performance analysis with the related methods. We conclude our work and discuss some future directions in Section 5.

2. Related Work

The real-time information generation nature of twitter makes it important to serve the situational information needs during emergencies. However, in addition to its short length characteristics, the informal user-generated tweets contain lots of unambiguous and unconventional word forms. Thus, it is challenging for the researchers to distill the correct information type accurately from tweet content.

To effectively utilize microblogging sites during disaster events, Rudra et al. [10] proposed a framework that first classified the tweets based on the consideration of the typicalities about disaster events and whether a tweet contains a mixture of situational and non-situational information. Later, the summarization technique was employed for better representation of the extracted situational information. Truong et al. [11] proposed a Bayesian approach to identify the informational tweets from the tweet streams, whereas Dutt et al. [12] proposed a system named SAVITR for extracting real-time location information from microblogs during an emergency situation or disaster period.

Moreover, Gosh et al. [13] introduced a task at the 2016 forum of information retrieval (FIRE) conference, in which the goal was to address the challenges of retrieving specific types of situational information from twitter posts during the disaster period. Basu et al. [14], [15] conducted a comparative performance evaluation of the traditional IR models for this task as well as analyzing the performance of the participants’ systems. Along with this direction, in the 2017 FIRE microblog track, Basu et al. [16] introduced a task to identify only the need tweets and availability type tweets from the tweet streams. The need and availability tweets are very important for coordinating relief operations in a disaster situation. Many teams have participated in these tasks with their proposed solution to tackle the challenges [16].

Recently, at the 2018 text retrieval conference (TREC), McCreadie et al. [3], [4] introduced an incident stream (TREC-IS) task which was focused on addressing the challenges of microblog retrieval during an emergency period. The main task of this track was to categorize the tweets related to an incident into different high-level information types.

Participants at the TREC-IS 2018 track [4], [5], [6], [7] utilized the manual feature engineering techniques with the classical learning algorithms including SVM, Naive-Bayes, random forest, decision tree, and K-nearest neighbors (KNN). In addition, some participants employed wordnet in their approaches [7], [17]. Other participants proposed deep learning based approaches including multilayer perceptron (MLP), convolutional neural networks (CNN), and long short-term memory (LSTM) [5], [6]. Few participants combined the classical method with deep learning models to improve classification accuracy [6].

However, most of the participants explored the traditional approaches in their proposed methods, which motivate us to address the actionable informative tweet classification problem defined in TREC-IS using state-of-the-art deep learning techniques in a unified architecture.

3. Proposed Framework

In this section, we describe the details of our proposed framework. Given a query related to an incident and a set of tweets, the goal of our proposed framework is to categorize the actionable informative tweets into the different high-level information types. For tweets related to disasters, examples of some high-level information types include GoodsServices, SearchAndRescue, Donations, MultimediaShare, Sentiment, Volunteer, Factoid, InformationWanted, Official, etc. The overview of our proposed framework is depicted in Fig. 1.

At first, our system fetches a query and the corresponding tweet set as a single batch and indexes them for further processing. Next, a naive rule-based classifier is applied to classify the tweets into the corresponding high-level information types. For the tweets that are not classified by the rule-based classifier, we consider the prediction label from our proposed neural network model. Finally, the set of labeled tweets returns to the user.
Intuitively, in disaster scenarios, it might be beneficial if we can define some rules to distill the tweet information quickly and accurately without employing extensive feature extraction procedures for neural network based methods. We choose the neural network based method over the traditional bag-of-words (BoW) based methods because tweets have length constraints and do not provide sufficient word occurrences. Besides, BoW based models have some strong limitations to handle the informal user-generated content, especially tweets. First, tweets contain the rare and noisy words incessantly, which leads to a severe feature dimensionality problem. Second, BoW considers all words of a document as independent, therefore failed to capture the word-order information. Third, words in the tweets are frequently polysemous i.e., a word may have several meanings and BoW failed to address polysemy problems [18], [19]. From this observation, we combine the rule-based classifier with a unified neural network model in our proposed framework.

3.1 Rule-Based Classifier

Rule-based classification schemes have been popularly used in various classification tasks because they are easy to design, understand, interpret, and classify the new samples quickly and effectively [20], [21], [22], [23]. Moreover, performance of the rule-based classifiers are comparable to decision trees [24], [25].

In rule-based classifiers, we usually construct a set of rules that determine a certain combination of patterns, which are most likely to be related to the different classes or information types. Each rule consists of an antecedent part and a consequent part. The antecedent part corresponds to the condition and the consequent part corresponds to a class label [26]. We can define a rule as follows:

\[ R_j : \text{if } x_1 \text{ is } A_{j1} \text{ and/or } \ldots \ldots x_n \text{ is } A_{jm} \]

then \( \text{Class} = C_j, \quad j = 1, \ldots, N \)

where \( R_j \) is a rule label, \( j \) is a rule index, \( A_{ji} \) is an antecedent set, \( C_j \) is a consequent class, and \( N \) is the total number of rules.

Our unsupervised rule-based classifier casts the TREC incident streams (TREC-IS) task as a multi-class classification problem and labels each tweet to the corresponding information types assigned by the rules. To achieve this, we define a set of rules based on the tweets’ language and indicator terms available within a tweet. Descriptions of our defined rules are presented in the subsequent subsections.

3.1.1 Language Related Rule

Even though twitter is a multilingual microblogging platform, we only consider English tweets as relevant in this research. Therefore, we define a rule based on the language of a tweet that is, if the language of a tweet is not English, we classify the tweet as Irrelevant information type. We employ a publicly available language detection library*1 to identify the language of a tweet.

3.1.2 Indicator Terms based Rule

A tweet may contain some highly influential indicator terms related to an information type that may be useful to categorize the tweet into the corresponding type. In this regard, we exploit the indicator terms of several information types and empirically define two rules for the two information types. One for the MultimediaShare category and the other for the Donations category. We prepare two curated lexicons of indicator terms for these categories. Examples of some highly influential indicator terms for MultimediaShare category include “#photos”, “video”, “@youtube”, etc. and some examples for Donations category include “donate”, “donations”, “fundraiser”, etc. If a tweet contains words from these lexicons, we classify it to the corresponding information type. The priority of the information type is determined by the number of lexicon words that the tweet contains.

3.2 Proposed Neural Network Model

In this section, we describe the details of our proposed neural network model that classifies the actionable informative tweets into different information types. Figure 2 depicts an overview of our proposed neural architecture.

At first, we utilize the multiple kernels based convolution filters with four different kernel sizes including (2, 3, 4, 5) to extract the higher-level feature sequences from the tweet embeddings. The generated feature sequences are then concatenated and fed into the nested LSTMs (NLSTMs) to learn long-term dependencies. To adopt the strength of transfer learning, we employ a pre-trained sentence embeddings model, InferSent to encode each tweet into a 4096-dimensional feature vector. In addition, we extract 19 hand-crafted features broadly grouped into four different categories including lexical and content relevant features, incident and event related features, sentiment aware features, and twitter specific features. Both the transfer learning

---

*1 https://code.google.com/p/language-detection/
features and hand-crafted features are then combined and sent to a multi-layer perceptron (MLP) module. Finally, representations from NLSTMs module and MLP modules are concatenated and sent to the fully-connected information type prediction module to determine the final label. In the following sections, we describe each of these components elaborately.

3.2.1 Embedding Layer
In the embedding layer, we represent each tweet based on the distributed vector representation of words that help the deep learning models to achieve better performances[27], [28], [29]. Such vector representation captures the semantics and contextual information in their values. To obtain the high-quality distributed vector representations, we utilize a pre-trained word embedding model in our proposed framework. The dimensionality of the word vector matrix defined as \( L \times D \), where \( L \) denotes the length of the tweet, and \( D \) denotes the word-vector dimension.

3.2.2 Convolution using Multiple Kernels (CMK)

The objective of using multiple kernel-based filters in the convolution layer is to obtain different types of effective features [30]. Previous studies already demonstrated the efficacy of using multiple kernels over the single one [30], [31]. Therefore, we perform the convolution operation that uses multiple kernels (i.e., the size of the convolution filters): 2, 3, 4, and 5 to extract the high-level features from the embedding matrix obtained from the previous stage. After performing the max-pooling operation, the univariate feature maps generated from each kernel are concatenated and passed to the next layer.

3.2.3 Nested LSTMs

Nowadays, long short-term memory (LSTM) based deep learning models are the most popular choice for sequential tasks. In our neural architecture, we employ the state-of-the-art nested LSTMs (NLSTMs) [9] model that achieved significant improvement to learn longer term dependencies compared to the single-layer or stacked LSTM model. In NLSTMs, the LSTM memory cells selectively read and write necessary long-term information through accessing their inner memory. Though LSTM is employing \( c_{t|t-1}^{outer} = f_{t} c_{t-1} + i_{t} g_{t} \), to estimate its outer memory cell value, NLSTMs use the concatenation \( f_{t} \odot c_{t-1}, h_{t} \odot g_{t} \) as an input to an inner LSTM (or NLSTM) memory cell, and set \( c_{t|t}^{outer} = h_{t}^{inner} \).

Figure 3 depicts an illustration of computational graphs of the LSTM and NLSTM. Such selective access to inner memories helps the NLSTMs to operate on longer time-scales and capture the contextual information effectively.

3.2.4 External Tweet Encoder

We exploit the convolutional-nested-LSTMS module to learn the context of the tweets. However, due to the complex nature of disaster related tweets and shortage of training samples, it might be beneficial if we incorporate knowledge from external sources to extract some generic information from tweets. We employ two different approaches to encode each tweet to extract the meaningful information as features. The first one is the transfer learning based features extractor via universal sentence embedding and the second one is the hand-crafted features extractor. Next, we describe each of the approaches in detail.

Transfer Learning Features: Disasters and other crises events are usually complex by nature and have some unique characteristics. That is why the type of events, affected people and the region, time, and many other factors have an impact on events news and information evolves in social media. Therefore, it is cumbersome to prepare effective training samples and extract relevant contextual information from the unstructured and noisy tweet contents. To overcome these limitations, one plausible solution is to transfer generic knowledge from existing tasks as an additional input to the designed model.

In transfer learning, a neural network model is trained on a dataset before being used as a feature extractor on the other dataset [32], [33]. The extracted transfer learning features are usually generic and applicable to other applications. This way enables a model to transfer knowledge which augments the learning process in a different target domain. We choose transfer learning via pre-trained sentence embedding due to its effectiveness over the word level embeddings [34], [35]. However, prior work on providing pre-trained sentence embedding models can be rami-fied into two types [36]: i) models that require fine-tuning according to the transferring task [37], [38], [39], [40] and ii) models that provide universal sentence embeddings and can be employed without fine-tuning the encoder parameters [34], [41], [42]. We follow the second type and employ a pre-trained generic sentence embedding model InferSent [34] without fine-tuning to extract the effective transfer learning features.

InferSent [34] is a universal sentence embedding model that embeds a full sentence into a vector representation. Such representation captures important features from a sentence since they utilize the inherit properties from their underlying word embeddings. In the InferSent model, a BiLSTM network with max-pooling function is employed to encode each sentence. It is trained on supervised data of the Stanford natural language inference (SNLI) dataset. InferSent model exploits the semantic nature of the SNLI task for obtaining generic sentence representations. Experimental results on several transfer tasks demonstrated that the semantic representations of sentences provided by this method are effective compared to the other popular unsupervised methods e.g., SkipThought vectors [43]. This observation motivates us to utilize this sentence embedding model to extract the effective transfer learning features. In our architecture, we employ the InferSent model trained on fastText [29] vectors to encode each tweet into a 4096-dimensional feature vector.

Hand-Crafted Features: We extract a set of 19 hand-crafted features broadly grouped into 4 different categories including
Table 1 List of hand-crafted features used in this work [6], [26].

| Type                        | Feature Definition                                                                 |
|-----------------------------|------------------------------------------------------------------------------------|
| **Lexical and Content Relevant** |                                                                                   |
| (1) TF-IDF [53] similarity score between an incident query and a tweet.   |                                                                                   |
| (2) Okapi BM25 [54] similarity score between an incident query and a tweet. |                                                                                   |
| (3) Language model with Dirichlet smoothing [55] score between an incident query and a tweet. |                                                                                   |
| (4) Tweet Length Feature: Number of words available in a tweet.             |                                                                                   |
| (5) Average Word Length Feature: Average length of the words available in a tweet. |                                                                                   |
| **Incident and Event Related** |                                                                                   |
| (1) Location Count Feature: Number of location names available in a tweet.   |                                                                                   |
| (2) Organization Count Feature: Number of organization names available in a tweet. |                                                                                   |
| (3) Person Count Feature: Number of person-information available in a tweet. |                                                                                   |
| (4) Noun Count Feature: Number of noun POS available in a tweet.             |                                                                                   |
| (5) Phone Number Count Feature: Number of phone numbers available in a tweet. |                                                                                   |
| (6) Known Already Count Feature: In an incident event, the number of previously posted tweets that are nearly identical (70% similar based on Cosine Similarity) with the corresponding tweet. |                                                                                   |
| **Sentiment Aware**                                                                                              |
| (1) Sentiment Polarity Feature: A binary feature that is assigned to 1 if a tweet has positive or negative sentiment polarity and 0 otherwise. |                                                                                   |
| (2) Positive Word Count Feature: Number of positive words available in a tweet based on the lexicon. |                                                                                   |
| (3) Negative Word Count Feature: Number of negative words available in a tweet based on the lexicon. |                                                                                   |
| (4) Emoticon Count Feature: Number of emoticons available in a tweet.       |                                                                                   |
| **Twitter Specific**                                                                                              |
| (1) HashTag Feature: A binary feature that is assigned to 1 if a tweet contains a hash tag and 0 otherwise. |                                                                                   |
| (2) HashTag Count Feature: Number of hashtags available in a tweet.          |                                                                                   |
| (3) URL Feature: A binary feature that is assigned to 1 if a tweet contains a URL and 0 otherwise. |                                                                                   |
| (4) Retweet Feature: A binary feature that is assigned to 1 if a tweet is a retweet of the other tweet and 0 otherwise. |                                                                                   |
| **Total**                                                                                                          |
| 19 Features                                                             |                                                                                   |

The first 3 lexical and content relevance features are used to estimate the lexical similarity between a given incident query and a tweet. In this regard, we generate the incident query by combining the query title and narrative. The intuition behind these features is that a tweet might contain more relevant and meaningful information if it has the highest similarity with the query. Besides, we extract the tweet length feature and average word length feature since a longer tweet might contain more information about the incident.

We also extract 6 incident and event related features that seem to be important during the disaster period. We utilize the Stanford named entity recognizer (NER) tool [44] to identify the location, organization, and person information to extract the corresponding features. Along with this direction, a publicly available library emoji4j [4] is utilized to estimate the phone number count feature. We also use the CMU ARK part-of-speech (POS) tagger [45] to identify the noun POS of each tokenized word which is required to extract the noun count feature. To estimate the known already count feature of a tweet, we enumerate the number of previously posted similar tweets. Two tweets are considered as similar if they are 70% similar based on the cosine similarity score.

Since most of the tweets posted in a disaster or emergency situation are usually sentiment sensitive, we extract four sentiment aware features to distill the sentiment dimension of a tweet. To estimate the sentiment polarity of a tweet, we use a publicly available package SentiStrength [46]. We construct the positive and negative sentiment bearing word lexicons as described in Ref. [47], where seven publicly available sentiment lexicons are used including the Bing Liu lexicon [48], subjectivity clues from [49], EffectWordNet [50], WordStat Sentiment Dictionary v3, NRC emotion lexicon [51], SentiStrength lexicon [46], and SentiWordnet [52]. We utilize these lexicons to estimate the lexicon based sentiment aware features. For the emoticon count feature, we use a publicly available library emoji4j [4] to identify the emoticon.

Moreover, twitter has some special characteristics including hashtag, retweet, and URL. A hashtag is a metadata tag and people usually use it to highlight the trending topics and events. Since disaster events usually reached the trending topic in twitter within the shortest span of time, therefore hashtag might be an important indicator to distill the tweets’ information. We extract two hashtag related twitter specific features as described in Table 1. To overcome the limitations of twitter length constraints, people use URL in their posted tweets to share extra information. Besides, an informative tweet might be retweeted by other users. We also exploit these important characteristics to extract the corresponding features as defined in Table 1.

3.2.5 Multilayer Perceptron (MLP)

After extracting transfer learning features and hand-crafted features, we concatenate them and pass to a fully connected multilayer perceptron (MLP) [56] network. An MLP is a feed-forward artificial neural network model that maps sets of input data that are passing through multiple fully connected hidden layers to generate the appropriate outputs. Multiple hidden layers help the model to learn the required information for identifying correct information types of a tweet effectively. MLP utilizes stochastic gradient descent based back-propagation algorithm [57], a supervised technique to learn all the parameters of the model.

3.2.6 Information Type Prediction and Model Training

After getting the final tweet representation from both the CMK+NLSTMs module and MLP module, we concatenate them and pass to a fully connected softmax layer for information type classification. The output of the softmax layer is the probability distribution over all the information type categories and the category with the highest value is assigned as the final label. In our model, cross-entropy is used as the loss function and the model is trained by minimizing the error defined as follows:

2 https://github.com/google/libphonenumber

3 http://provalisresearch.com/Download/WSD.zip

4 https://github.com/ktihota/emoji4j
\[ P(n^{(i)}, n^{(j)}) = \sum_{j=1}^{k} 1[n^{(i)} = j] \log(n_j^{(i)}) \]

where \(m^{(i)}\) is the training sample with its true label \(n^{(i)}\). \(n_j^{(i)}\) is the estimated probability in [0, 1] for each label \(j\). \(1[\text{condition}]\) is an indicator which is 1 if true and 0 otherwise.

We adopt the stochastic gradient descent (SGD) algorithm to learn the model parameter and use the Adam optimizer \([58], [59]\). To tackle the overfitting problem and prevent the complex co-adaptations that arise due to a small set of training samples, we employ the commonly used dropout \([60]\) and L2 weight regularization \([61]\) techniques in our proposed neural architecture.

3.3 Combining the Classifiers

After developing the rule-based classifier and our proposed neural network-based classifier, we combine them to classify the tweet into the different high-level information types. At first, the rule-based classifier is applied to classify the tweet to the corresponding information type. For tweets that are not classified by the rule-based classifier, we consider the prediction label from our proposed neural network model as the final label.

4. Experiments and Evaluation

4.1 Dataset Collection

To validate the effectiveness of our proposed method, we made use of the TREC incident streams (TREC-IS) benchmark dataset \([3], [4]\) released at the TREC-2018. The dataset contains 21 query topics, where the training set contains 6 query topics and the test set contains 15 query topics. The topics are selected from six different incident event types including wildfire, earthquake, flood, tornado/hurricane/hurricane, bombing, and shooting. Each query topic is composed of topic_number, query_title, query_type, Wikipedia_url, and query_narratives. For each query topic, a set of relevant tweets is sampled from twitter. TREC-IS organizers provided 3771 training tweets related to 6 training query topics, where 1335 tweets were annotated with the label and corresponding indicator terms. Though we employed a naive rule-based classifier at the entry stage of our proposed approach, we used the full training dataset to train our proposed neural network model. In addition, from the training data, the indicator terms of the MultimediaShare category and Donations category were used to create the lexicons described in Section 3.1.2. Similarly, among the provided 22238 test tweets related to 15 test query topics, 19784 tweets were labeled by the TREC assessors. Systems were evaluated based on the labeled tweets.

The organizers also provided an ontology of information types, which contains 25 high-level information types or class label broadly grouped into Request, Report, CallToAction, and Other categories. Table 2 summarizes the overall statistics of the 2018 TREC-IS dataset and the statistics of the annotated tweets distribution among the 25 high-level information types in both the training and test set are presented in Table 3.

4.2 Dataset Preprocessing

The data preprocessing stage is initiated with tokenization. To distill more information from a tweet text, we employed a hashtag segmentation technique and replaced the hashtag with the segmented words. To achieve this, we utilized a tool provided by Baziotis et al. \([62]\), where they used the Viterbi algorithm \([63]\) to select the probable segmented words based on the word statistics from large twitter corpora. We also removed the non-English characters and symbols from the tweets. In twitter, people often used non-standard word forms to express their thought informally and concisely. For example, sometimes people use “fireeeeeee” instead of “fire,” “floooood” instead of “flood”, “trnadoo” instead “tornado”, and “hlp” instead of “help”. We employed two lexical normalization dictionaries collected from Refs. \([64]\) and \([65]\) to obtain the original form of such unconventional words. Besides, stopwords play a negative role and eventually degrade the performance of the classifier system because they do not carry any action-oriented information. For stopword removal, we applied the Indri’s standard stoplist \(^{65}\).

4.3 Evaluation Measures

To evaluate the performance of our proposed method, we applied the evaluation measures used in the 2018 TREC-IS track \([3], [4]\), where participants’ systems were tasked with identifying one most relevant information type per-tweet. But during the ground truth generation of the test set, human assessors were allowed to select as many information types as appropriate for a single tweet. Therefore, the organizer used two ways referred

\(^{65}\) https://www.lemurproject.org/stopwords/stoplist.dft

| Table 2 | The statistics of 2018 TREC-IS dataset. |
|----------|---------------------------------------|
| **Category** | **Train** | **Test** |
| Number of Incident Events | 6 | 15 |
| Number of Annotated Tweets | 1335 | 19784 |
| Number of High-level Information Types | 23 | 25 |
| Average Number of Tweets Per Event | 267 | 1318 |
| Average Number of Tweets Per Information Type | 53 | 1736 |

| Table 3 | Tweets distribution among high-level information types. |
|----------|---------------------------------------|
| **Intent** | **High-level Information Type** | **Train** | **Test** |
| CallToAction | CallToAction-Donations | 15 | 804 |
| | CallToAction-MovePeople | 26 | 27 |
| | CallToAction-Volunteer | 2 | 116 |
| Request | Request-GoodsServices | 0 | 126 |
| | Request-InformationWanted | 10 | 172 |
| | Request-SearchAndRescue | 2 | 286 |
| Report | Report-CleanUp | 2 | 62 |
| | Report-EmergingThreats | 36 | 686 |
| | Report-Factoid | 140 | 2383 |
| | Report-FirstPartyObservation | 28 | 1325 |
| | Report-Hashtags | 4 | 3363 |
| | Report-MultimediaShare | 127 | 3974 |
| | Report-Official | 52 | 403 |
| | Report-ServiceAvailable | 15 | 1076 |
| | Report-SignificantEventChange | 34 | 415 |
| | Report-ThirdPartyObservation | 15 | 3807 |
| | Report-Weather | 42 | 4160 |
| Other | Other-Advice | 39 | 1209 |
| | Other-ContinuingNews | 251 | 4871 |
| | Other-Discussion | 51 | 2060 |
| | Other-Irrelevant | 163 | 2605 |
| | Other-KnownAlready | 113 | 1191 |
| | Other-PastNews | 12 | 1351 |
| | Other-Sentiment | 132 | 6952 |
| | Other-Unknown | 26 | 77 |
to as multi-type and any-type evaluation criteria to evaluate the performance.

In the multi-type evaluation metrics, there are two sub-types: **Positive Class**, where the categorization performance per information type is estimated in a 1 vs. All manner and **Overall**, that estimates the overall classification accuracy. A system gets a full score if it identifies all the categories that the human assessor selected for that tweet. In any-type evaluation, a system is considered to correctly categorize a tweet and gets a full score if it identifies any of the categories that the human assessor selected for that tweet.

Since participants’ systems were tasked with identifying one most relevant information type per-tweet, the 2018 TREC-IS organizers employed the any-type evaluation criteria to evaluate the performance of a system. This evaluation measure is useful to evaluate the absolute performance of a model. Four standard evaluation metrics including micro average precision, recall, F1 score, and accuracy were used to estimate the performance of a system, where the micro average F1 score was considered as the primary evaluation measure. We also report the comparative results based on the multi-type evaluation, though system performance cannot be evaluated perfectly under this measure. However, multi-type evaluation is primarily useful for comparative performance analysis between information types and between events. We utilized the evaluation scripts provided by the TREC-IS organizers for ensuring appropriate evaluation and comparison.

To check whether there is a statistically significant difference between the performances of the two classification systems, we used the McNemar’s test\(^6\), \(^7\), \(^8\) at a 95% confidence level (\(p\)-value < 0.05). McNemar’s test is a statistical test that applies to a \(2 \times 2\) contingency table as presented in Table 4. It is also known as McNemar’s Chi-Square test because sampling distribution of the McNemar statistic uses the chi-square\(^9\) distribution.

The null hypothesis of this test is that the performances of the two classifiers are equal (i.e., they have the same error rate). The critical value for the McNemar statistic at a 95% confidence level is 3.84. The null hypothesis is rejected if the McNemar statistic > 3.84 (when \(p\)-value < 0.05).

### 4.4 Model Configuration

In this section, we describe the hyper-parameters settings of our proposed neural model. Our model was designed on Tensorflow\(^{10}\) and trained on a GPU\(^{11}\) to utilize the efficiency of parallel computation of tensors. A simple random grid search with 10-fold cross-validation on the full training dataset is employed to select the optimal hyper-parameters. The hyper-parameters search space is illustrated in Table 5.

We employed the 300-dimensional fastText embedding model pre-trained on Wikipedia with skip-gram\(^{29}\) to initialize the word embeddings in the embedding layer described in Section 3.2.1. In our multiple kernel-based convolutions (CMK) described in Section 3.2.2, we employed 4 different kernel sizes including (2, 3, 4, 5) and the number of filters was set to 600. In our model, the nested LSTM modules contain 2 layers and the MLP module contains 3 layers. We applied 150 epochs to train our model with a batch size of 32 and an initial learning rate was set to 0.001 with the Adam optimizer. Both the NLSTMs layers and the MLP layers were dropped out with a probability of 0.02. L2 regularization with a factor of 0.01 was applied to the weights in the softmax layer. To get a stable and reproducible performance of our proposed neural model, we set up a random seed using numpy.random.seed (45).

However, to keep the generalizability of the InferSent sentence embeddings and due to the sparse distribution of tweets across information types, we did not fine-tune the InferSent\(^{34}\) model. We employed the pre-trained InferSent\(^{34}\) model as a generic transfer learning features extractor. Unless otherwise stated, default settings were used for the other parameters.

### 4.5 Results and Analysis

We now evaluate the actionable tweet categorization performance of our proposed method in this section. Following the 2018 TREC-IS\(^{4}\) benchmark, we consider the any-type evaluation criteria to estimate the performance. The summarized results of our proposed method based on different experimental settings are presented in Table 6.

At first, we report the results based on the baseline. We used the n-gram based bag-of-words (BoW) feature with the scikit-learn\(^{56}\) implementation of linear support vector machine (SVM) classifier, LinearSVC and multinomial naive Bayes (NB) classifier, MultinomialNB as our baseline systems. In our experiments, we consider the word n-grams (1-, 2-, and 3-gram) features with the TF-IDF weighting scheme and regularization parameter, \(C=10\) for LinearSVC. Default settings were used for the other parameters. We also incorporate our proposed rule-based classifier with the SVM and NB settings to make an effective comparison with our proposed neural approach.

Next, we report the overall results of our proposed method. Experimental results show that our proposed method outperforms the above mentioned baselines by at least 5.14% (SVM+Rule-based) and at best 8.99% (NB) in terms of the primary evaluation.
Metric F1 score. This deduces the effectiveness of our method over the traditional BoW based approaches.

To evaluate the effectiveness of each component, we performed the component ablation study on our proposed model. In this regard, we removed one component each time and repeated the experiment. From Table 6, it can be observed that when removing the individual major components including rule-based classifier (Cf.), convolution using multiple kernels (CMK), nested LSTMs (NLSTMs), and multilayer perceptron (MLP) the results decreased by 1.36%, 5.14%, 2.12%, and 2.16%, respectively, in terms of primary evaluation measure micro avg. F1 score. It shows that multi-kernel convolution (CMK) has the highest impact on the performances. This validates our rationale for using different kernel sizes in the convolution layer to capture different types of feature abstractions. However, the NLSTMs and MLP components have a similar kind of impact. This deduces the importance of the NLSTMs module to learn longer-term dependencies and the MLP module to incorporate additional effective information through transfer learning and hand-crafted features. We also see that our rule-based classifier has the least impact on the overall performance. This is because we used very few rules for identifying information types.

From the analysis of individual components contribution in our model, we have seen that CMK has the highest contribution. Therefore, it is expected that the (CMK+NLSTMs) module might also have a significant contribution to our model. When removing (CMK+NLSTMs) module from our model, we have observed a drastic decrease in performance, which is 28.26% in terms of micro avg. F1 score. This deduces the effectiveness of (CMK+NLSTMs) to capture the tweet contexts effectively. Besides, the aggregation of MLP module imputed with the hand-crafted and transfer learning features, enhance the performance of our proposed neural model to identify the correct information type of the tweets. We have also seen that when removing (CMK+MLP) and (NLSTMs+MLP) components, the results decreased by 4.95% and 1.58%, respectively. This deduces the combined contribution of these components.

However, from the experimental results, it seems that we achieved the perfect recall when removing our rule-based classifier i.e. when incorporating our rule-based classifier the recall rate is slightly dropped though the rate of other evaluation measures are improved. This is because in our rule-based classifier particularly the language-related rule misjudges some positive tweets. We have identified two plausible reasons regarding the misjudgment: the human assessor judged some non-English tweets as relevant during the ground-truth generation and our used language detection tool failed to decide the language of some tweets due to the noisy words or very few word occurrences.

Moreover, we performed the McNemar’s statistical test (p-value < 0.05) to validate whether the decrease in performance is significant or not in the ablation study and validate the improvement related to the baselines. From the results, we have seen that when removing each component from our model the decrease in performance is significant. This deduces the importance of each of the components in our neural architecture. We also obtained a significant difference in results against the baselines.

### 4.6 Comparison with Related Work

To evaluate the performance of our method against the current state-of-the-art, we compared the performance with the top-performing teams at the 2018 TREC-IS task [4] named as cbnuS2 [5], KDEIS4_DM [6], umdhcilfasttext [8], cbnuS1 [5], NHK_run2 [7], and uogTr_R3_asp [4]. The comparative results are presented in Table 7. The results showed that our proposed method achieved a 5.97% improvement over the top-performing system cbnuS2 [5] in terms of primary evaluation measure micro avg. F1 score based on any-type evaluation criteria, which validates the effectiveness of our method. We also obtained the best result in terms of the precision and accuracy compared to the participants’ systems.

However, we also reported the results based on multi-type evaluation criteria despite its limitation for overall system performance estimation according to the 2018 TREC-IS [4] benchmark. Under the multi-type positive class metrics, performance per information type is estimated in a 1 vs. All manner (considering only true positives and true negatives). A system only gets the full score if it selects all the assigned categories by the human assessor. But the 2018 TREC-IS task the participants’ systems to provide only one category per tweet while the assessors provide multiple categories per tweet. Therefore, performance evaluation under this metric cannot be perfect. For instance, if the ground truth of the tweet contains 4 information types, a system only receives a maximum of a 1/4 score for that tweet. In addition, as shown in Table 3, we have seen that some information types have very few training samples, therefore our proposed method did not capture the context of these information types and gave preference to the other information types. The scenario also illustrated in Fig. 4, where we see that our proposed method did not classify any tweets to the few categories. Therefore, the scores of these information types affect the macro average evaluation measures precision, recall, and F1 score which in turn drop down.
the performance of our model compared to some related methods including cbnuS2 and cbnuS1 [5].

In contrast, the overall accuracy score in multi-type evaluation criteria is much higher compared to the positive class score. Since a tweet belongs to more than one category, therefore true negatives are vastly more common than true positives. That is why overall accuracy is much higher compared to the positive class scores. Our proposed method obtained slightly better results under this evaluation metric.

In addition, McNemar’s statistical testing (p-value < 0.05) was conducted for performance comparison with other methods. From Table 7, the results showed that a high degree of significant differences was observed between our proposed method and other related methods. We also compared the performance of our model with the participant’s median as shown in Table 7.

cbnuS1 and cbnuS2 [5] utilized the conceptual representation of tweets to train the SVM classifier. For the conceptual representation of tweets, the cbnuS1 system considered the terms, event entities, category indicator entities, and information type entities. In addition to these entities, the cbnuS2 system utilized the URL entities and user entities. Since the tweets are short in length and contain the rare and noisy words incessantly, considering only the entity information might not be sufficient to represent the tweet context. However, in our proposed method we overcome these limitations by employing state-of-the-art deep learning techniques and transfer learning features for effective tweet representation. KDEIS4_DM [6] employed a rule-based classifier with a linear weighted combination of the hand-crafted feature based SVM classifier and DeepMoji [72] based neural network model. However, DeepMoji is trained on emotion-related tweets and designed specifically for emotion-related text modeling. Therefore, features generated from DeepMoji may not represent the context of all types of tweets effectively and only a linear weighted combination with other classifiers may not reduce its biases towards emotional tweets. In contrast, our method is free from such biases and focused on all types of tweets equally. NHK_run2 [7] applied a bag-of-words (BoW) based multilayer perceptron (MLP) model. However, BoW features have the limitations of capturing word-order information and addressing the challenges of polysemy (i.e. different words have similar meaning) information. In contrast, we employed the deep semantic representation of tweets to mitigate this problem. umdhcilfasttext [8] trained a fastText [29] model based on their collected tweets to generate the feature vectors and employed the Naive-Bayes classifier for classification. This model only focused on capturing the semantic meaning of individual words using fastText and didn’t employ any methods e.g., CNN, LSTM to learn the compositionality of words therefore didn’t represent the context of the tweet effectively. In contrast, our convolutional nested LSTM module mitigates this issue and generates effective tweet representation. uogTr_R3_asp [4] used the weighted combination of three modules based on indicator terms, information type dictionary, and FactFinder (a series of machine-learned classification models) to predict the information type categories, where highest weight was given to the information type dictionary module and didn’t employ any methods e.g., CNN, LSTM to learn the context of the tweet.

Fig. 4 (Multi-type) Performance comparison among the high-level information types.

| Method                          | Multi-type (Macro Avg.) | Any-type (Macro Avg.) | McNemar’s Test on Any-type Evaluation Criteria (p-value) |
|---------------------------------|-------------------------|-----------------------|--------------------------------------------------------|
| Proposed Method                 | 0.2161 0.0986 0.1162    | 0.9095               |                                                        |
| Positive Class (1 vs. All)      | 0.4559 0.7780 0.5749    | 0.4213               |                                                        |
| Overall Accuracy                | 0.2187 0.1164 0.9048    | 0.9035               |                                                        |
| Top Performing Teams in TRECIS-2018 [4] | 0.2104 0.1005 0.1187 | 0.9042               |                                                        |
| cbnuS2 [5]                      | 0.2159 0.0945 0.1059    | 0.9095               |                                                        |
| KDEIS4_DM [6]                   | 0.4534 0.7260 0.5582    | 0.4022               |                                                        |
| umdhcilfasttext [8]             | 0.4472 0.7402 0.5575    | 0.4064               |                                                        |
| cbnuS1 [5]                      | 0.4483 0.7143 0.5509    | 0.3997               |                                                        |
| NHK_run2 [7]                    | 0.3136 1.0000 0.4775    | 0.3136               |                                                        |
| uogTr_R3_asp [4]                | 0.3978 0.6165 0.4775    | 0.3385               | N/A                                                    |
| Participant Median              | 0.1827 0.0784 0.0825    | 0.8993               |                                                        |

Table 7 Comparative performance (Precision, Recall, F1 Score, and Accuracy; higher is better) of our method against the state-of-the-art on TREC-IS 2018 test set. The best results are highlighted in boldface. * indicates the statistically significant difference between our proposed method and the other methods; (McNemar’s Test: p-value < 0.05).
Table 8: Training and test tweets distribution among high-level information types across events.

| Events | Number of Sample | 2012 | 2013 | 2014 | 2015 | 2016 |
|--------|------------------|------|------|------|------|------|
| Number | 283              | 247  | 235  | 244  | 162  | 164  |
| Tweet Distribution among High-level Information Type | | | | | | |
| Donations | 10 0 7 1 0 1 | | | | | |
| MovePeople | 0 1 9 1 0 5 | | | | | |
| Volunteer | 1 0 0 0 0 0 | | | | | |
| Emergency | 0 0 0 0 0 0 | | | | | |
| Volunteers | 0 0 0 0 0 0 | | | | | |
| InformationWanted | 1 1 3 3 1 | | | | | |
| SearchAvail | 0 0 0 0 0 0 | | | | | |
| Weather | 1 0 0 0 0 0 | | | | | |
| EmergingEvents | 3 2 2 2 2 2 | | | | | |
| Illinates | 133 22 | | | | | |
| Illinates | 3 9 3 10 | | | | | |
| Illinates | 0 0 0 0 0 0 | | | | | |
| Illinates | 152 21 | | | | | |
| Illinates | 0 1 1 1 1 | | | | | |
| Illinates | 11 3 4 12 | | | | | |
| Illinates | 1 1 1 1 1 | | | | | |
| Illinates | 6 2 10 | | | | | |
| Illinates | 1 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
| Illinates | 1 | | | | | |
### Table 9: Examples of successful and unsuccessful tweets based on our proposed methods and three related methods including SVM+Rule-based, NB+Rule-based, and 2018 TREC-IS top performing system cbnuS2 [5]. The tweets are related to various incident events and boldfaced labels are one of the correct labels for the respective tweets.

| Event          | Tweet Samples                                                                 | SVM+Rule-based | NB+Rule-based | cbnuS2 [5] | Proposed Method  |
|----------------|-------------------------------------------------------------------------------|----------------|---------------|------------|-----------------|
| 2012 Guatemala Earthquake | At least 48 killed as 7.4 mag earthquake strikes Guatemala http://t.co/sZCeOufGf (via @news360upp) | KnownAlready   | KnownAlready  | KnownAlready | ContinuingNews  |
| 2012 Italy Earthquakes | 5.8-magnitude quake hits northern Italy, causes more buildings to crumble: http://t.co/WTdGQ0OT - VW via @AP | KnownAlready   | KnownAlready  | KnownAlready | ContinuingNews  |
| 2012 Philippines Floods | @Tabenimitch stay tuned for announcements. Check the #ReliefPH relief ph | Donations      | ContinuingNews | Sentiment   | Advice          |
| 2013 Alberta Floods | The floods in Calgary are crazy  | PastNews        | ContinuingNews | MultimediaShare | Sentiment       |
| 2013 Australia Bushfire | RT @NSWFRS: #Lithgow Fire: The fire has jumped the Limes Line of Road. #NSWFRS #fire #lithgow | EmergingThreats | ContinuingNews | Irrelevant   | FirstPartyObservation |
| 2013 Boston Bombings | RT @WHLive: Happening at 11ET: President Obama speaks at an interfaith service dedicated to victims of the bombing in Boston. http://t.co/... | Factoid        | Sentiment     | Factoid     | Official        |
| 2013 Manila Floods | RT @MMID: FLOOD UPDATES: as of 10:45 AM, (MANILA) Recto Moraya - gutter deep, Rizal Recto - 1/2 tire deep. Taft Kalaw (cont) http://t.co/5? | Advice         | ContinuingNews | MultimediaShare | Factoid        |
| 2013 Queensland Floods | Whoohoos! Thankyou Energex power back on Kiels Mtn 4559. Time for a cuppa. #bigwet big wet | Factoid        | ContinuingNews | Irrelevant   | Sentiment       |
| 2013 Typhoon Yolanda | RT @ABCBCNchannel2: Through his hectic schedule, he finds time and ways to help the #Philippines. Thank you so much @justinbieber :) philipinnes. | Donations      | ContinuingNews | Discussion   | Sentiment       |
| 2011 Joplin Tornado | Tornado warning O.O what tha fawk? Akron area -_- this world is deadening: http://yearbook.com/a/1/cfz2hs | Discussion     | ContinuingNews | SignificantEventChange | Sentiment       |
| 2014 Chile Earthquake | Chile’s Earthquake status as of April 3, 2014 GMT+8 http://t.co/aQbBvWkJR | KnownAlready   | KnownAlready  | KnownAlready | FirstPartyObservation |
| 2014 Typhoon Hagupit | @CruzRojasEup Did you see #typhoon #Hagupit with #MeteoEarth? http://t.co/vtCsurjdp typhoonhagupit meteo earth? | Sentiment      | ContinuingNews | Irrelevant   | Weather         |
| 2015 Nepal Earthquake | @WelshToy I just searched Nepal climate change and that is actually scary. | FirstPartyObservation | ContinuingNews | Irrelevant   | Discussion      |
| 2018 FL School Shooting | School shooting plot suspect kept journal of plans https://t.co/tqGQxnMmce | Factoid        | ContinuingNews | ContinuingNews | PastNews        |
| 2015 Paris Attacks | A #French police union official tells AP there were 2 suicide attacks and one bombing near stadium. #ParisAttacks: french paris attacks. | ContinuingNews | ContinuingNews | ContinuingNews | ThirdPartyObservation |

| Event          | Tweet Samples                                                                 | SVM+Rule-based | NB+Rule-based | cbnuS2 [5] | Proposed Method  |
|----------------|-------------------------------------------------------------------------------|----------------|---------------|------------|-----------------|
| 2012 Guatemala Earthquake | RT @BBCBreaking: Update: #Guatemala’s President Molina says at least 48 people killed by #earthquake http://t.co/BcZnyWZI guatemala’s earthquake | Factoid        | ContinuingNews | ContinuingNews | Official        |
| 2013 Alberta Floods | Calgary floods trigger an oil spill and a mass evacuation: http://t.co/04MxW7ED | ContinuingNews | ContinuingNews | ContinuingNews | MovePeople      |
| 2013 Australia Bushfire | ‘Mega-fire’ fears in Australia: http://t.co/0QzqWyXwr | ThirdPartyObservation | ContinuingNews | ContinuingNews | Sentiment       |
| 2013 Boston Bombings | RT EhabZ: Boston suspect’s twitter Dzhokhar, he doesn’t seem like a terrorist. | ContinuingNews | ContinuingNews | Irrelevant   | Discussion      |
| 2013 Typhoon Yolanda | Typhoon YOLANDA Archive — Tropical Cyclone Warning for Shi... Typhoon YOLANDA Trop... http://t.co/3Nkt34KLa | Official       | ContinuingNews | Official     | Weather         |
| 2015 Paris Attacks | Shocking Paris attacks leaves 60 dead, others held hostage https://t.co/QBrTQHwfrw | Factoid        | Factoid        | Factoid     | ContinuingNews  |
| 2018 FL School Shooting | RT @WPXI: UPDATE: Suspect in Florida school shooting still at-large, officials say. Look for developing information on Channel 11 News at 5? | ContinuingNews | ContinuingNews | ContinuingNews | Discussion      |
largest number of tweets belonging to the Other-Sentiment information type. For some other information types including CallToAction-Donations, Other-ContinuingNews, Report-Weather, Report-MultimediaShare, Report-Factoid, Other-Advice, and Other-Discussion, our method classified a moderate number of tweets. However, our method did not classify any tweets to the seven information types. Since there are multiple labels annotated to each tweet and our system annotated one category per tweet according to the 2018 TREC-IS benchmark, therefore our system prioritized other information types over these types. To identify the reason, we observed the tweet distribution across information types and events illustrated in Table 3 and Table 8, respectively. Our investigation revealed that no training sample provided for the Request-GoodsServices and Request-SearchAndRescue information types and very few (2 to 4) training samples provided for the Report-CleanUp, CallToAction-Volunteer, and Report-Hashtags types, which makes it difficult for our model to learn the contextual information effectively for those categories.

Similarly, the incident event-wise performance of our method illustrated in Fig. 5. We observed that our method obtained a similar kind of performance for different types of events which are broadly grouped into earthquake, tornado/typhoon/hurricane, bombings, shootings, floods, and wildfires. From Table 3 and Table 8, we observed that the major information types which are shared across events including Sentiment, ContinuiningNews, Weather, MultimediaShare, ThirdPartyObservation, and Factoid. Figure 4 shows that our proposed method captures the trends of these information types but ThirdPartyObservation, which is in turn helps our model to obtain a nearly similar performance across events as illustrated in Fig. 5. Therefore, we can deduce that our method is not biased towards specific types of events thus is easily generalizable to other events.

For qualitative analysis, we have enlisted successful and unsuccessful example tweets of various events classified by our proposed method and three related methods including SVM+Rule-based, NB+Rule-based, and the top-performing system of the 2018 TREC-IS track cbnuS2 [5] in Table 9. Boldfaced labels are one of the correct labels for the respective tweets.

From the illustration of successful examples, we see that our proposed method learned the context of the tweet effectively compared to the other methods and classify the tweet to the correct label. Considering the example tweets “@sabenimitch stay tuned for announcements. Check the #ReliefPH relief phi” and “The floods in Calgary are crazy 🇨🇦”, we see that our method classifies them to the Advice and Sentiment information types, whereas other methods failed to categorize these tweets to any of the correct labels. This deduces the effectiveness of our method over the other methods.

Besides, we also enlist some unsuccessful examples to analyze the shortcomings of our method. We see that some tweets are ambiguously and/or minimally labeled by the human assessors i.e. these tweets did not contain all the correct labels. For example, it seems that the tweet “Calgary floods trigger an oil spill and a mass evacuation: http://t.co/uW4NMxW7ED” contains information about people’s evacuation, therefore MovePeople should be one of the categories for this tweet. However, the ground truth did not contain this label. The same things happened for the “‘Mega-fire’ fears in Australia http://t.co/6RqeqYsWxr” tweets. Though there is a strong indication of the Sentiment category, the ground truth of this tweet did not contain this label. However, there is still room for improvement of our method. Since the dataset does not provide much training data and imbalanced across classes and events as shown in Table 3 and Table 8, effective data augmentation might help our model to further improve the performance.

5. Conclusion and Future Directions

In this paper, we have proposed a neural network model with a naive rule-based classifier for the actionable informative tweet categorization. We trained a multilayer perceptron (MLP) focused on transfer learning features and hand-crafted features. In addition, upon extracting higher-level feature sequences through multiple kernels based convolution, we employed the nested LSTMs for learning long-term dependencies. The generated feature sequences from both modules are then concatenated and passed to a fully connected layer to estimate the final tweet label. Experimental results on the 2018 TREC-IS dataset have shown that our proposed neural model learned the contextual information effectively which in turn improved the actionable informative tweet categorization performance and exceeded the current state-of-the-art methods by a large margin.

In the future, we have a plan to exploit the automatic data augmentation, location information, temporal information, retweet, follower count, and author-follower relation to improve the classification performance. Moreover, we intend to generalize and evaluate our model for identifying multiple information types, because a single tweet might contain the information of more than one type.

Acknowledgments The part of this research is supported by MEXT KAKENHI, Grant-in-Aid for Scientific Research (B), Grant Number 17H01746.

References

[1] Sakaki, T., Okazaki, M. and Matsuo, Y.: Tweet analysis for real-time event detection and earthquake reporting system development, IEEE Transactions on Knowledge and Data Engineering (TKDE), Vol.25, No.4, pp.919–931 (2013).
[2] Ounis, I., Macdonald, C., Lin, J. and Soboroff, I.: Overview of the TREC-2011 microblog track, Proc. 20th Text REtrieval Conference (TREC), NIST (2011).
[3] McCreadie, R., Buntain, C. and Soboroff, I.: Guidelines v1.0 - TREC 2018 incident streams track (2018).
[4] McCreadie, R., Buntain, C. and Soboroff, I.: TREC incident streams: Finding actionable information on social media, Proc. 16th International Conference on Information Systems for Crisis Response and Management (ISCRAM) (2019).
[5] Choi, W., Jo, S.-H. and Lee, K.-S.: CBNU at TREC 2018 incident streams track, Proc. 27th Text REtrieval Conference (TREC), NIST (2018).
[6] Chy, A.N., Siddiqua, U.A. and Aono, M.: Neural networks and support vector machine based approach for classifying tweets by information types at TREC 2018 incident streams task, Proc. 27th Text REtrieval Conference (TREC), NIST (2018).
[7] Miyazaki, T., Makino, K., Takei, Y., Okamoto, H. and Goto, J.: NHK STRL at TREC 2018 incident streams track, Proc. 27th Text REtrieval Conference (TREC), NIST (2018).
[8] Buntain, C.: Learning information types in social media for crises, Proc. 27th Text REtrieval Conference (TREC), NIST (2018).
[9] Moniz, J.R.A. and Krueger, D.: Nested LSTMs, Asian Conference on Machine Learning (ACML), pp.530–544, Springer (2017).
Owens, J.D., Rajaraman, A. and Ullman, J.D.: **Mining of massive datasets**, Cambridge University Press (2014).

Robertson, S.E., Walker, S., Beaufort, M. and Willett, P.: Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive track, *Proc. 7th Text REtrieval Conference (TREC-7)*, NIST Special Publication 500-242, pp.253–264, National Institute of Standards & Technology (1999).

Ponte, J.M. and Croft, W.B.: A language modeling approach to information retrieval, *Proc. 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.275–281, ACM (1998).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research (JMLR)*, Vol.12, pp.2825–2830 (2011).

LeCun, Y.A., Bottou, L., Orr, G.B. and Müller, K.-R.: Efficient back-prop, *Neural Networks: Tricks of the Trade*, pp.9–48, Springer (2012).

Kingma, D.P. and Ba, J.: Adam: A Method for Stochastic Optimisation, *Proc. 3rd International Conference on Learning Representations ICLR* (2015).

Tieleman, T. and Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural Networks for Machine Learning, Vol.4, No.2, pp.26–31 (2012).

Kingma, D.P. and Ba, J.: Adam: A Method for Stochastic Optimisation, *Proc. 3rd International Conference on Learning Representations ICLR* (2015).

Tieleman, T. and Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural Networks for Machine Learning, Vol.4, No.2, pp.26–31 (2012).

Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.: Improving neural networks by preventing co-adaptation of feature detectors, *CoRR*, Vol. abs/1207.0580, pp.1–18 (2012).

Tieleman, T. and Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural Networks for Machine Learning, Vol.4, No.2, pp.26–31 (2012).

Advani, A., and Walker, B.F.: The Chi square test: An introduction, *COMMSIG review*, Vol.4, No.3, p.61 (1995).

Advani, A., and Walker, B.F.: The Chi square test: An introduction, *COMMSIG review*, Vol.4, No.3, p.61 (1995).

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: Tensorflow: A system for large-scale machine learning, *Proc. 12th USENIX Conference on Operating Systems Design and Implementation (OSDI)*, pp.265–283, USENIX Association (2016).

Upadhyay, S., Faruqui, M., Dyer, C. and Roth, D.: Cross-lingual Model for social media language, *Proc. 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp.50–58 (2012).

Kingma, D.P. and Ba, J.: Adam: A Method for Stochastic Optimisation, *Proc. 3rd International Conference on Learning Representations ICLR* (2015).

Advani, A., and Walker, B.F.: The Chi square test: An introduction, *COMMSIG review*, Vol.4, No.3, p.61 (1995).

Advani, A., and Walker, B.F.: The Chi square test: An introduction, *COMMSIG review*, Vol.4, No.3, p.61 (1995).

Abu Nowshed Chy received his B.Sc. (Hons.) degree from the University of Chittagong, Chittagong, Bangladesh in 2012 and his M.Eng. and Ph.D. degrees from the Toyohashi University of Technology, Toyohashi, Japan in 2016 and 2019. He is currently working as a lecturer at the Graduate School of Computer Science and Engineering Department, University of Chittagong, Bangladesh. His research interests include microblog search, crisis informatics, temporal information retrieval, multimodal information retrieval, opinion mining, natural language processing, and deep learning.

Umme Aymun Siddiqua received her B.Sc. (Engg.) degree from the International Islamic University Chittagong, Chittagong, Bangladesh in 2016 and her M.Eng. degree from the Toyohashi University of Technology, Toyohashi, Japan in 2019. She is currently working as an instructor at the Asian University for Women, Bangladesh. Her research interests include opinion mining and knowledge discovery from social media data especially stance detection, sentiment analysis, emotion analysis, and hate speech detection.

Masaki Aono received his B.S. and M.S. degrees from the Department of Information Science from the University of Tokyo, Tokyo, Japan, and his Ph.D. degree from the Department of Computer Science at Rensselaer Polytechnic Institute, New York. He was with the IBM Tokyo Research Laboratory from 1984 to 2003. He is currently a professor at the Graduate School of Computer Science and Engineering Department, Toyohashi University of Technology. His research interests include text and data mining for massive streaming data, and information retrieval for multimedia including 2D images, videos, and 3D shape models. He is a member of the ACM and IEEE Computer Society. He has been a Japanese delegate of the ISO/IEC JTC1 SC24 Standard Committee since 1996.