Expert Systems With Applications 195 (2022) 116562

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Topic2Labels: A framework to annotate and classify the social media data through LDA topics and deep learning models for crisis response

Junaid Abdul Wahid [a], Lei Shi [b,*], Yufei Gao [b,*], Bei Yang [a], Lin Wei [b], Yongcai Tao [a], Shabir Hussain [a], Muhammad Ayoub [c], Imam Yagoub [a]

[a] School of Information Engineering, Zhengzhou University, 450001, China
[b] School of Cyber Science and Engineering, Zhengzhou University, 450000, China
[c] School of Computer Science and Engineering, Central South University, Hunan, China

ARTICLE INFO

Keywords:
Social media
Natural language processing
Neural network
Topic modeling
Annotation
Classification
Transformer
Crisis response

ABSTRACT

The abundant use of social media impacts every aspect of life, including crisis management. Disaster management needs real-time data to be used in machine learning and deep learning models to aid their decision making. Mostly the data that is newly generated from social media is unstructured and unlabeled. Current text classification models based on supervised deep learning models heavily rely on human-labeled data that very small size and imbalanced in the context of disasters, ultimately affecting the generalization of models. In this study, we propose Topic2labels (T2L) framework which provides an automated way of labeling the data through LDA (latent dirichlet allocation) topic modeling approach and utilize Bert (the bidirectional encoder representation from transformer) embeddings for construction of feature vector to be employed to classify the data contextually. Our framework consists of three layers. In the first layer, we adopt LDA to generate the topics from the data, and develop a new algorithm to rank the topics, and map the highest ranked dominant topic into label to annotate the data. In the second layer, we transform the labeled text into feature representation through Bert embeddings and in the third layer we leveraged deep learning models as classifiers to classify the textual data into multiple categories. Experimental results on crisis-related datasets show that our framework performs better in terms of classification performance and yields improvement as compared to other baseline approaches.

1. Introduction

During the crisis, people spread information on social media and share related, real time and valuable information. People tweet about asking for help, expressing their situations, and offering help to others. Social media such as twitter has become a dominant platform for organizations and people to collect and disseminate information during natural crises such as floods, earthquakes, wildfires and recently COVID-19 pandemic (Nguyen et al., 2017; Vieweg, 2012). Situational information like relevant texts can be greatly enhanced by generated text in the form of tweets (Vieweg, 2012). Rapid analysis of messages posted on micro-blogging platforms such as Twitter can assist humanitarian organizations such as United Nations can gain situational awareness, learn about needs of affected people, medical emergencies, and donations related information, etc. at different locations. Tweets contain rich information which can be used by these organizations in response to people needs (Nguyen et al., 2017). Social media tweet classification is a text classification task that aimed at identifying whether a tweet text contains information related to crises or not. For example, tweet "Update: 3 die in Australia flood waters since the weekend" is tweet about update of dead persons in Australia floods and it is related to crisis, while tweet "Australian government announced export business policy to boost the exports" is totally irrelevant in the context of floods. The main objective of text classification for crisis response is to identify/classify if a tweet is related to specific type of defined situational information categories. However, current text classification models suffers from lack of labeled data in real time which prevents them from reaching a generalized model (Caragea et al., 2016; Li et al., 2018). Beside this, it is not feasible to manually annotate a large amount of tweets in real-time for crisis events (ALRashdi & O'Keefe, 2020). Automatic classification of tweets to extract relevant

* Corresponding authors.
E-mail addresses: junaid.a.wahid@gs.zzu.edu.cn (J.A. Wahid), shilei@zzu.edu.cn (L. Shi), yfgao@zzu.edu.cn (Y. Gao), iebyang@zzu.edu.cn (B. Yang), weilin@zzu.edu.cn (L. Wei), ieyctao@zzu.edu.cn (Y. Tao), shabir@gs.zzu.edu.cn (S. Hussain), 214718006@csu.edu.cn (M. Ayoub), imamkeela@gs.zzu.edu.cn (I. Yagoub).

https://doi.org/10.1016/j.eswa.2022.116562
Received 19 August 2021; Received in revised form 30 December 2021; Accepted 17 January 2022
Available online 9 February 2022
0957-4174/© 2022 Elsevier Ltd. All rights reserved.

text remains a challenging task, because: (1) tweets are short — only 280 characters, consequently, it is hard to extract useful information without enough context; (2) they often contain slang languages, abbreviations, which are ambiguous. Classifying tweet into different categories post further difficulties such as inappropriate labels and the semantic ambiguity in the language makes it hard to understand the context of tweets. In accordance with all these inherent difficulties in classifying tweets, computer cannot generally agree on annotator's rate, on which annotators agree with each other (Nguyen et al., 2017) on labels. Therefore, despite much advancement in natural language processing (NLP), automatic annotation of text and better classification of texts remain problems that need to be investigated.

For contextual classification of textual data specifically crisis related text, supervised learning approaches need well labeled data for training, But in real time, during the crisis there is scarcity of context related labeled data for training, Later the labeled data arrive in small batches after it is labeled by paid workers and organizations. It is not feasible to train the model from scratch whenever the new labeled text arrives due to the velocity of new textual data, thus it arises the need of an automated annotation approach. Beside properly labeled data, text classification also needs contextual representation of texts through feature vectors or word embeddings. Traditional supervised machine learning and deep learning methods utilized feature engineering schemes like TF–IDF, bag of words feature scheme or internal embedding layers for conversion of texts into feature vectors or feature embeddings that feed into classifiers (Imran et al., 2015), but these feature vector schemes ignore the context of words within the sentence or ultimately within documents, that leads to application of contextual word embeddings schemes while training supervised classifiers. In the context of crisis, due to a lot of variation in data, adapting the model to train the labeled data and features vector or word embedding schemes is often not feasible.

Because of this, approaches that automatically annotate texts and deep learning methods with the fusion of embedding schemes are desirable to contextually classify the text. An automated approach to annotate and classify the tweet texts will enhance the accuracy of classifiers and reduce the cost of manually labeling the text. Our main objective is to investigate, for the first time the application of topic modeling approach to annotate the textual data and utilizing transformer-based contextual Bert based embeddings with various deep learning classifiers to identifying various type of relevant texts in crisis situation for crisis response and improve the classification performance. Therefore, our main contributions are as follows:

- We proposed an automated approach to annotate textual data by utilizing LDA distributions thus, reduce the manual label cost.
- For extracting dominant LDA topic, we have developed new topic ranking algorithm (T-TF-IDF) to rank LDA topic distributions.
- We utilized and trained deep learning methods with transformer based Bert embedding layer for contextual classification of data.
- In comparison with other baseline approaches, we evaluate our framework from the perspective of classification measures and statistical measures and reached significant results.

The rest of the paper is organized as follows: the following section covers the related work on automatic labeling and classification. Next we proposed our framework in detail with different subsections, then we discuss results and analysis of different classifiers. Furthermore, we carry out the discussion, and in the end, we conclude our paper with conclusion and future work section.

2. Related work

2.1. Automatic annotation of text

Various research studies utilized different techniques to automatically annotate the text. Researchers in Go et al. (2009) use distant supervision approach to automatically label the disaster-related tweets, they have utilized an external knowledge base from the related context to label the tweets, and from that knowledge base they extracted the crisis related keywords and rank those keywords to label the tweets. They further mentioned that keywords play a vital role in identifying disaster related tweets of different categories in crisis situations. In addition, distant supervision techniques with different variations have been successfully employed to label twitter data with different approaches. In Kralj Novak et al. (2015), authors assumed that emotions in tweets express the feelings of people, on the basis of this assumption they utilized the emotions to label the tweets for sentiment classification task, such as, if the tweet text contains happy emoticons, then it is labeled as a positive tweet, and if it contains negative, angry, or sad emoticon then it labeled as negative tweet. The study by Mohammed et al. (2017) employed distant supervision for topic classification task, in which they transfer labels from tweets of topically focused accounts to tweets posted by general Twitter accounts. The tweet to link approach used by researchers in Magdy et al. (2015) they leveraged YouTube xs to assign labels to tweets that contain links to those videos. Researchers in Athira et al. (2021b) utilized already labeled less amount of posts to label the data, in their approach, they first developed the base classifier with the 1000 labeled posts, utilized that classifier to classify the remaining data in the context of medical text and, then to check the efficiency of newly labeled data they evaluate the classification results of classifiers in terms of evaluation measures such as accuracy, precision, recall and F1 score. Automatic annotation of tweets is done by researchers in Athira et al. (2021a) by utilizing enhanced bootstrapping algorithm comprises of LSA (latent semantic analysis) and word2vec, they assumed both these model as distributed semantic models. A recent study (Krommyda et al., 2021) on the annotation of tweets used Plutchik's eight basic emotions to annotate tweets into eight categories, emoji's, keywords, and semantic relationships are used to identify in an objective way the emotion expressed in a short text, after they annotate the dataset into multiple categories they evaluate the accuracy using LSTM (long short term memory) model as a classifier. Utilization of pre-existing annotated corpora as a training set is most commonly used approach to annotate the text with same domains, such as researchers in de Carvalho et al. (2020) analyze six annotated datasets of Brazilian and Portuguese languages, and used one of them as a training dataset, for further annotation of text, this approach can also beneficial in a way that texts other than the English language are hard to label and their lexicons are hard to find. Sometimes the context is also important in accurately classifying the data, and researchers in Menini et al. (2021), they utilized the previously labeled data and manually re-annotated that text into specified contextual categories for detection of abusive text, then to determine the abuse detection in a contextual way they applied Bert end to end model for classification.

2.2. Classification of text

In prior studies, many classifications method have been utilized to classify the textual data, and recently in learning based approaches, deep learning methods outperform traditional machine learning methods in classification while processing larger datasets. Researchers in Alrashdi and O'Keefe (2020) used Bidirectional Long Short-Term Memory (Bi-LSTM) with a maximum pooling and global vector embeddings as an embedding scheme for classification of crisis tweets. In Nguyen et al. (2017), researchers utilized a convolutional neural network (CNN) to classify tweet texts based on their relatedness to give crisis and based on information type. For the embedding schemes they have utilized word2vec and crisis embeddings, and they trained this crisis domain-specific embeddings by themselves with a vocabulary size of 20 million. To analyze topics of discussion in online health communities to study the patient's experience, researchers in Athira et al. (2021a), have applied several machine learning and deep learning classifiers to detect

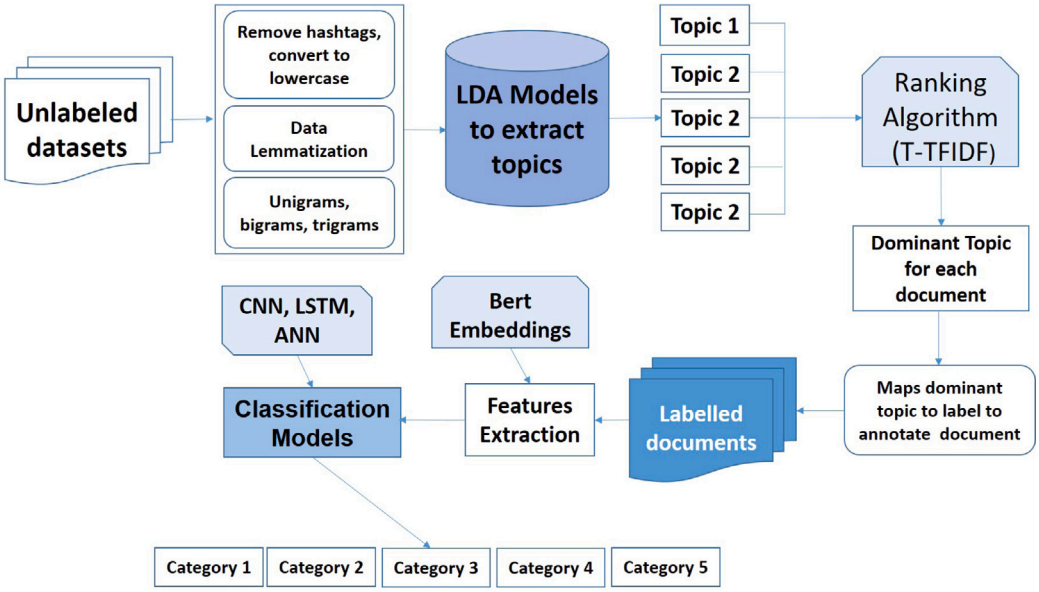


Fig. 1. Proposed framework detail with each component.

the topics of discussion. They employed deep-learning based CNN, Long Short-Term Memory (LSTM), and BiLSTM classifiers with Bert embedding techniques, machine learning based SVM (support vector machine) and KNN (K-nearest neighbor) classifiers with Bert embeddings for classification of online health community texts. In another study, researchers in Menini et al. (2021) used Bert for embeddings and as a classifier (end to end) both for classifying the contextual abusive text. They have manually annotated the abusive text, and check the efficiency of Bert model because Bert said to be the model to accurately classify the text in a specific context. in Madichetty and Sridevi (2020), this study utilized the image based features along with textual features to classify the tweet for disaster response. For text feature extraction they used CNN and for classification they used ANN (artificial neural network), and for image based features they utilized fine-tuned VGG-16 with CNN to extract image features from image and classify an image in tweet, then combining the output of text-based features and image based features to further classify the tweet text.

3. Proposed framework

It is essential to label datasets and use contextual embeddings with suitable classification models to achieve desired classification performance, for this purpose, we first apply the automated labeling technique through LDA topic distributions, then, we develop new ranking algorithm to rank those topics for dominant topic extraction, then extracted features by using Bert embeddings, and then used widely used deep learning models with Bert embeddings for text classification. Our proposed framework is shown in Fig. 1, in which we showed various steps that we will discuss in detail in this section.

3.1. Topic Modelling

Latent Dirichlet Allocation (LDA) was first introduced by Blei et al. (2003), it is a probabilistic generative model that can be used to estimate the multinomial observations by an unsupervised learning approach. With respect to model the topics, it is a method to perform latent semantic analysis (LSA). The main idea behind LSA is to extract the latent structure of topics or concepts from the given documents. Initially, the term latent semantic coined by Kim et al. (2020) showed that the co-occurrence of words in the documents could show the semantic structure of the document and ultimately find the concept or topic. With LDA, each document is represented as multinomial distribution of topics where topic can be seen as high level concepts to documents. The assumption on which is based is that document is a collection created from topics, where each topic is presented with a mixture of words.

From the above model depicted in Fig. 2, the generalization of LDA is described as follows:

For each document sample m∈M topic proportions θm from the alpha dirichlet distribution, Then for each word placeholder n in the document m:

1. We randomly Choose a topic Z_{m},n in accordance with proportions of a sample topic.
2. We randomly choose a word W_{m},n from the set of multinomial distributions ϕk of already chosen topic.

In the generalization process of LDA, the α and β are hyper vector parameters that determine the dirichlet prior on θ, m is a collection of topic distributions for all the documents and on parameter ϕ, they determine the word distributions per topic (Pavlinek & Podgorelec, 2017). Whereas, M: total no of documents, N: total no of words, K: is number of topics, ϕk :word distributions of topic K, Z_{m},n:a document topic over words, W_{m},n:topic words of specific document and, θm:topic distribution of document.

Data pre-processing

All the pre-processing steps are shown in Fig. 3, like removing punctuations, transforming to lowercase letters, and make into lists, the detail of the remaining steps is in following subsections.

3.1.1. Tokenization and lemmatization

Tokenization is the process of breaking the document or tweets into words called tokens. A token is an individual part of a sentence having some semantic values. Like Sentence "hurricane is coming" would be tokenized into 'hurricane', 'is', 'coming'. We have utilized the Spacy function with core English language model for tokenization and lemmatization.[1] The beauty of this Spacy function is that it gives you part of speech detail of every sentence, and can choose from that which part of speech need for further processing in the specific context. Spacy is capable enough to also give sentence dependencies in case need them while

[1] https://spacy.io/usage/models.

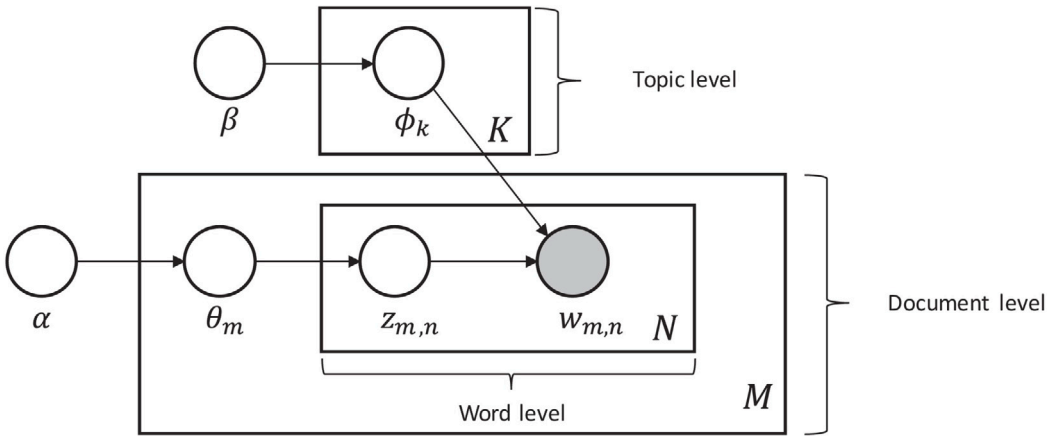


Fig. 2. Generalization of LDA topic modeling model.

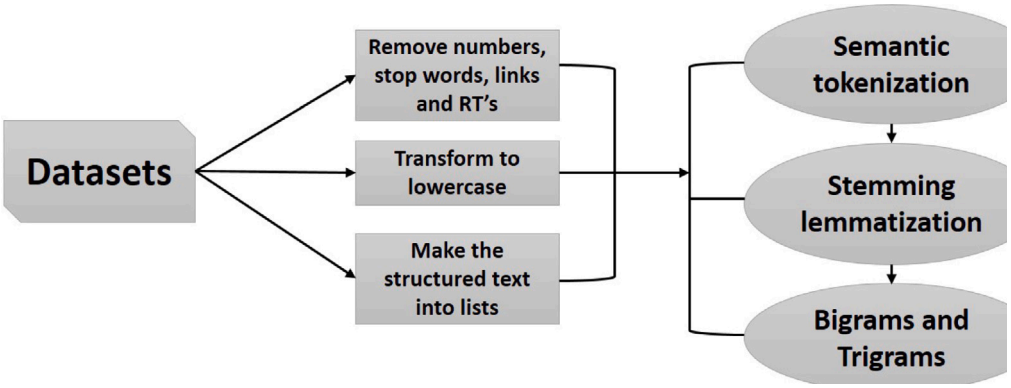


Fig. 3. Pre-processing steps involved in data pre-processing.

performing graph embedding's. After tokenization, we need to see which part of sentence we need and also need to extract the words into their original forms. Both the lemmatization process and stemming process are widely used for this purpose. Many typical text classification techniques use stemming with the help of port stemmer, and snowball stemmer, words 'compute', 'computer', 'computing', 'computed' would be reduced into word 'comput', and a slight drawback with stemming is that it reduces the word into its root form without looking into that the word is found in dictionary of that specific language or not, as you can see 'comput' is not a dictionary word. There comes the lemmatization, and with Spacy we performed the lemmatization. Lemmatization also reduced the word into its root form but by keeping mind the dictionary database. With lemmatization the above examples of words ('compute','computer','computed','computing') would be reduced to root form as ('compute','computer','computed','computing') respectively by keeping in mind the dictionary. While implementing the Lemmatization part we keep the sentence with words having only the 'nouns', 'adjectives', and 'verbs', that is helpful if you need to be more specified about the LDA topics.

3.1.2. Bigrams and trigrams

Sometimes in large and sparse texts, we see the nouns or adjectives are combination of multiple words, therefore, to make the semantic context of words into sentences we need bigrams and even trigrams so that it will not break into single separate unigram tokens and lost the meaning and semantic of a sentence. Bigrams is an approach to making words that is of two tokens to remain into its semantic shape so that sentence contextual meaning would not be lost. We achieved this through Genism's phrases class[2] that allows us to group semantically related phrases into one token for LDA implementation, such as ice_cream, new_york listed as single tokens. The output of genism's phrases bigrams mod class is a list of lists where each one represents reviews, documents or tweets and strings in each list is a mix of unigrams and bigrams. In the same way, for the sake of uniformity of three phrases words tokens, we applied trigrams through genism's phrases class to group semantically related phrases into single tokens for LDA implementation. This is normally applicable on country names such as united _states_of_america. The output of genism's phrases trigrams is a list of list where each list represents review, document or tweets and strings in the list and it is a mix of uni-grams, bigrams and trigrams.

3.2. LDA model

In topic modeling section we have discussed the LDA topic model architecture with components, here we will see how we extract the topics, then determine the number of topics and after that, how those topics converted into labels to annotate the tweets. For the accurate annotation of datasets meaningful topics need to be extracted, therefore, we applied every hyper-parameter and investigate measures score such as coherence and prevalence to extract the suitable, relevant and optimal number of topics. To apply the LDA model there is a specific representation of the content that we need in the form of corpus and along with a different we need the dictionary that assist that corpus. For different LDA models, we create different types of corpus, with unigrams, bigrams and trigrams. LDA model is specifically described in detail in 2.

[2] https://radimrehurek.com/gensim/models/ldamodel.html.

3.2.1. LDA model selection

LDA model selection to determine optimal number of topics was the most challenging task, as it can ultimately impacts the annotation and results of supervised classifiers. Therefore, to choose the best LDA model along with number of topics in the model was a time consuming task. To find the exact number of topics suited for better LDA model was the main focus of previous studies (Greene et al., 2014). There are evaluation measures to determine the performance of best LDA models to define optimal number of topics. These evaluation measures are coherence scores and perplexity scores, and the coherence measure scores a single topic by measuring the degree of semantic similarity between high scoring words in topics. These measurements help distinguish between topics that are semantically understandable and those topics that are components of statistical inference (Mutanga & Abayomi, 2020). Perplexity is the measure of how well a model predicts a sample and is measured as the normalized log-likelihood of a held out test set. A lower perplexity score and higher coherence score means a better generalization of LDA model. But focusing on the log-likelihood part, you can think of the perplexity metric as measuring how probable some new unseen data is given the LDA model that we learn. However, recent studies show that predictive likelihood such as perplexity and human judgments are not correlated (Chang et al., 2009). This indicates that optimizing for perplexity may not yield human interpretable topics, and it leads us to focus on only coherence scores to check the performance of LDA models and find out the optimal number of topics.

3.2.2. Coherence score

The coherence score is for accessing the quality of learned topics through different LDA models. A set of statement of facts is said to be coherent if they support each other, thus, a coherent fact set can be interpreted in a context that covers all or most of the facts. In our context, a set of topics is said to be coherent in a document, if it depicts meanings that are representative of the document. Therefore, to determine the LDA model with optimal number of topics, we used coherence score as it is essential to find optimal number of topics, because these topics will ultimately convert into labels to use for annotating datasets. We have used the coherence measure C_V for all our LDA models. C_V measure is based on a sliding window, one-set segmentation of the top words and an indirect confirmation measure that uses normalized point wise mutual information (NPMI) and the cosine similarity.

$$CV = \sum i < j \quad \text{score}(w_i, w_j) \tag{1}$$

In Eq. (1), for one topic, the words i, j being scored in $\sum$ i < j score (w_i, w_j) have the highest probability of occurring for that topic. We just need to specify how many words in the topic to be consider for overall score.

3.2.3. Topic coherence analysis

Assessing the results shown in Table 1 with different topic numbers and different variants of bigrams and trigrams, we have found that on COVID-19 dataset it provides better results and coherent topics with the setting (bigrams with 5 topics). Most topics contain enough words occurrence to reveal useful information and do not contain too many words to make the topic unreadable. For the Disaster dataset, the setting (bigrams with 5 topics, bigrams with 10 topics) gives comprehensible results but bigrams with 5 topics marginally better with 10 topics, hence choosing 5 topics worked well for both our datasets.

Social media data is short, noisy and sparse, therefore, less number of topics seems to have a better performance than the LDA models with more number of topics. For both datasets coherence performance also seems better with the bigrams than trigrams. By investigating a coherence C_V values for each topic and word occurrence of each word in the topic, we explored that for less number of topics, each topic tends to have a better score and words in each topic have a more significant appearance in word occurrence, thus causing a higher coherence values. This can be described as no need of fewer number of words, for each topic, probability of belonging to a particular topic increases. It is because, social media text length has not so long, maximum characters are 280, in twitter, therefore, in this case, fewer topics and words in those topics best represent the documents.

Table 1
Coherence measure values of covid-19 and disaster datasets.

LDA models	Coherence (COVID-19)	Coherence (Disaster)
5 topics with bigrams	0.786	0.731
10 topics with bigrams	0.721	0.721
15 topics with bigrams	0.678	0.683
20 topics with bigrams	0.689	0.701
5 topics with trigrams	0.712	0.654
10 topics with trigrams	0.623	0.632
15 topics with trigrams	0.658	0.617
20 topics with trigrams	0.567	0.593

Also, while examining the topics, we found that topics with high coherence values with high-frequency words revealed relevant information about the topic. For example, COVID dataset (bigrams with 5 topics) has a highest coherence score of 0.786, In this setting, topic 2 revealed very handy and essential information with words such as “lockdown”, “mask”, ”quarantine”, “help” “lockdown” and “health”. Based on numerous examples like this and investigating coherence scores of each parameter setting, we conclude that, on social media datasets fewer topics and bigrams setting are the optimal settings for finalizing the number of topics. Next, after finding the optimal number of topics, we will extract one dominant topic from LDA model topics, for labeling of each tweet.

Information in the topics that are extracted from COVID-19 dataset seems to be relevant, keywords shown in Table 2 indicate that every topic contains specific information of different aspects. For example, topic 3 has words that depict information about the emotional support and help seeking during the pandemic, people talk about health care workers, giving emotional support to them, and talking about help seeking of different items needed during the pandemic. Similarly when you look at terms used in topic 0, it shows that this topic is about announcements of coronavirus, notification about the measures taken by the government and public health authorities, talks on medical equipment reserves and hospital conditions etc.

Similarly information extracted through topics from the disaster dataset is also relatable to different types of categories. As shown in Table 3, topic 3 contains information about death tolls, weather updates, about specific area which is Queensland and about specific type of crisis which is flood. Topic 1 shows the public reaction which they showed in their anger, attitudes, and feelings in a negative way towards government policies in crisis situations, therefore it categorizes as “People criticism in crisis situations”. Information about injured, people death counts, updated news about the crisis is all contained in topic 0, therefore we categorized it as News about deaths and injured people.

3.3. Dominant topic

After extracting the optimal number of topics from LDA model, we need to find the dominant topic to assign it to a document, every document has optimal number of topics, thus we need to find the dominant topic among the set of topics for a document, then assigns that topic as a label to a document. The intuition behind extracting the dominant is, when we apply the TF–IDF (term frequency–Inverse document frequency) on a set of documents, we are basically doing is comparing the importance of words between documents.

Therefore, we treat all the topics of a single document in a single category and apply TF–IDF, the result would be single topic per document and resultant score would be importance of words of topic in a

Table 2
Extracted topics keywords and description details: COVID-19 dataset.

Topics/classes	Keywords	Description
Topic 0	Trump, president, youtube, briefing, say, video, pm, cure, instruction, COVID, city, states, CDC	Announcements/Information from authorities, and government about policies
Topic 1	Mask, quarantinelife, wear, park, California, beach, vacation, saturday, store, COVID-19	Measures/Precautions need to be taken against the virus
Topic 2	Lockdown, get, day, time, think, spend, people, quarantine, COVID-19	Quarantine Life style during the lock down
Topic 3	Help, need, care, thank, health , work support, business, worker, COVID-19	Emotional support/Help seeking in emergency pandemic situation
Topic 4	Death, people, case, die, test, state, virus, number, report, coronavirus	No of cases and deaths of people, no of tests taken of people for coronavirus

Table 3
Extracted topics keywords and description details: disaster dataset.

Topics/classes	Keywords	Description
Topic 0	Explosion, plant, fertilizer, victim, prayer, tornado, family, affect injured, news, update, thought, affect	News about injured death and damaged affected by bombing
Topic 1	Texas, people, west, think, break, death, fire, suspect, happen, shit, fuck, camera, blast	People criticism in crisis
Topic 2	Help, rescue, thank, hope, miss, friend, donation, send, hurricane, get	Help seeking
Topic 3	Flood, Queensland, crisis, count, weather, GOD, emergency, north, toll, area, disaster	Update and help seeking in flood crisis
Topic 4	Water, rise, report, house, rain, home, kid, city, job, resident, mayor, evacuate, blame, hurricane	Announcement from government

document, and it will give us a dominant topic by comparing a topic's words probability distributions.

$$T - TF - IDF = \frac{t_i}{w_i} X \log \frac{m}{\sum n_i t_i} \tag{2}$$

Where T-TF-IDF = topic term frequency and inverse document frequency, and frequency of each word t is extracted for each topic (i) and divided by total number of words (w) of each topic (i). This action can be seen as a form of regularization of frequent words in the topic, next document m is divided by frequency of words across total topics n, t.

Algorithm 1: Extract dominant topic and maps into label to annotate the tweets

Input: unlabeled tweets corpus
Output: labelled tweets corpus with dominant topic
ni= Topic for each document;
T-TF-IDF= topic based variant of term frequency inverse document frequency;
Initialization;
ni ← LDA (un-labeled tweets);
word probabilities= word occurrence in specific topic;
extract dominant topic ← tweets corpus;
while *not the end of tweets file* **do**
  extract dominant_topic, topic_num;
  for *each tweet* **do**
    dominant_topic = T-TF-IDF(word probabilities);
    topic_num = dominant_topic index;
  end
  for *each tweet* **do**
    labeled_tweet ← dominant_topic (topic_num);
  end
  return labeled tweets;
  output= new file with tweets and its corresponding labels;
end

All the steps outlined in algorithm 1 are to extract dominant topic that maps into label to annotate the tweets. In the first step, we take unlabeled tweets as an input applies all the pre-processing steps to make the dataset suitable for further processing, then we apply LDA to extract different number of topics, it is to be noted that to extract the optimal number of topics we used the topic coherence score as a measure. After that, LDA gives the topics in the shape of different words with their occurrence probability score in the topic, In short, different words with probability scores makes a single topic. Uptil now we have topics with word probabilities, then we applied our proposed ranking algorithm called T-TF-IDF to extract the dominant topic from a list of number of topics for a single document, this process we have done for all the tweets in the dataset. After getting the dominant topic number for each tweet in the dataset, we referred this dominant topic as a label to annotate tweets and save all the tweets in a separate file, the resultant file would be tweets with labels in the form of dominant topic that were extracted from LDA and ranking algorithm. The LDA models with different number of topics and their coherence score comparisons is given in Section 3.2.3.

4. Training classification models

Most social media data is sparse, noisy and full of slangs, therefore, in order to feed data into classification models the textual data needs to be transform into feature vectors first. Mostly traditional supervised text classification models use traditional feature extraction methods such as TF–IDF, BOW for features extraction. However, not all sentences are simple, people on social media use different context. The problem of polysemy should be solved by considering the relationship between words before and after, therefore we use Bert pre-trained model to extract features to be used in classification algorithms.

4.1. Bert word embeddings

Bert is a deeply bidirectional, unsupervised language representation, pre-trained using only plain text corpus. It includes variants that use English Wikipedia with 2.5 million words. Unlike other context-free feature extraction models which generates a single word embedding

representation for each word in the vocabulary (Chatsiou, 2020; Devlin et al., 2018), Bert considers the context for each occurrence of a given word, providing a contextualized embedding that is different for each sentence. Given the nature of language modeling where future cannot be seen, previous models have been limited to two types of unidirectional representations (from left to right and right to left). Bert utilizes masked language models to predict random words and learn bidirectional representations. Bert also includes information from both directions instead of relying on only on direction (Naseem et al., 2020). Bert offers word representations that are dynamically informed by other words around them.

Considering the huge time and memory requirements of the Bert-large model, in our study we utilized the same parameter settings as the basic version of BERT in Devlin et al. (2018). We fine tuned the Bert based uncased model and set the number of (mask) self attention layers L and heads A to 12 with 12 encoder layers as well as feed forward networks with 768 hidden units extract transformer-based bidirectional contextualized representation of our texts.

4.2. Classification algorithms implementation

In order to classify big crisis related data, deep learning classification models are more suitable (Chatsiou, 2020), therefore, in next step of our study, we want to analyze the classification performance with different deep learning algorithms based on newly annotated dataset through our automated labeling approach (T2L). We have utilized deep learning algorithms suited for text classification such as CNN, LSTM and ANN. In next section we briefly describe the classifiers along with the implementation detail.

4.2.1. Artificial neural network classifier

Neural networks are represented by a network diagram that contains combinations of nodes connected by direct links. These nodes are structured in three different layers named, an input layer, hidden layer and output layer of nodes (Moraes et al., 2013). ANN also classified as feed forward network, and nodes are connected in on way direction. Every connection of node has weight, whose values are approximated by reducing an error function in a training process of gradient descent. An output value is calculated through two steps of a simple mathematical neuron model. At first, a model calculates a weighted sum of its inputs and then utilizes an activation function to this sum values to derive an output. In our study we have defined an output layer of 5 neurons because we are dealing with multiclass classification. The relu activation function used in neurons for the calculations, and Adam optimizer was used to find out the optimal value of each weight in neural network with 4 and 8 epochs.

4.2.2. Convolution neural network classifier

Researchers in Kim (2014) first outlined the idea of utilizing the CNN for text classification, then CNN has gained much of popularity in NLP specifically for text classification and achieved good results (Bilbao-Jayo & Almeida, 2018). We implement a CNN model for text classification with a single layer of convolution on top of the word embeddings that are extracted by utilizing Bert (Devlin et al., 2018). Tweets text as sequence of words then mapped into indexes after that into sequence of words. Vectors are then fed into CNN, we then implemented the convolution operation with 100 filters and three different filter sizes ($2 \times d$, $3 \times d$, and $4 \times d$). Each group of filters reduces dimensionality by utilizing the 1 max pooling, 0.5 dropout rate is applied to avoid over-fitting (Srivastava et al., 2014). In the end, fully connected layer with softmax function computes the probability values over the labels, in our case there are 5 labels. Also, we Performed optimization with Adam optimizer, after that we fit the model with 8 epochs and 1280 batch size, which give better results.

4.2.3. Long short term memory classifier

LSTM is a version of recurrent neural networks (RNN) used for text data sequential modeling. As compared to RNN, in LSTM only the most important data is passed to the next layer instead of whole data (Behera et al., 2021). Vanishing gradient problem or exploding is one of the main problem of RNN. LSTM was formulated to address the problem of exploding. LSTM would be able to possess a cell memory, which helps to retain the condition. One good reason to choose LSTM is that it is good for keeping important information. It has three gates, forgot gate, input gate and output gate, LSTM model able to solve the problem of gradient vanishing by adjusting the information in a cell state using these 3 gates (Jelodar et al., 2020). LSTM model contains different layers and it starts with the embedding layer that used to facilitate the learning of word embeddings, in our study, we have utilized Bert based word embeddings, then hidden layer contains LSTM layers, one can use many types of LSTM layers in hidden layers, we used one LSTM layer, then pooling layer is responsible for reducing the input from the perspective of spatial as well as facilitate regulating over fitting. This layer has two pooling types max pooling and average pooling, the values that are used in these two types are maximum value for the former and average value for the latter (Muhammad et al., 2021). Then fully connected layer outputs categories in the form of multi-dimensional array, in our study it has 5 categories.

We defined a sequential model and added various layers into it. The first one is embedding layers. It represents using Bert based word embeddings, then next layer is LSTM layer with 128 neurons, then LSTM layer followed by dropout layers with rate of 0.2 for regulating the network. In the end the final dense layer is the output layer with 5 cells representing categories along with softmax activation function. In order to get the optimal parameter for our model we use Adam as an optimizer and category cross entropy loss. In the end, our model with 6and 4 epochs and 64 batch size give better results.

4.3. Performance evaluation measures

To verify our proposed framework, we performed classification task on different datasets of different type of crises, one is disaster related dataset, and another one is COVID-19 pandemic dataset. Classification task was to classify the tweets into 5 different categories, we measure the classification scores different evaluation parameters, such as Accuracy, Precision, Recall, and F-measure.

Confusion matrix is a matrix mostly used in machine learning and deep learning classification problems to visualize the representation of statistical values obtained through experiments. It demonstrates the values of actual and predicted level of each tweet for the classifier (Behera et al., 2021). In our study we have created the 5X5 confusion matrix. In 5X5 matrix, we need to calculate the values for each class separately. The confusion matrix has following four components:

- True Positive (TP): It shows the actual value and predicted values are same. In our case the actual values is if category 1 classified as category 1.
- False Positive (FP): The false positive value for a class will be sum of corresponding column except for the TP value.
- True Negative (TN): It will be sum of values of all columns and rows except the values of a class that we are calculating the values for.
- False negative (FN): For a class, it will be sum of values of corresponding rows except for the TP value.

The performance of a classifier evaluated from the accuracy, precision, recall, and f-measure parameters. The equations of evaluation index are as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F1 - Measure = 2x\frac{Precision \quad x \quad Recall}{Precision \quad x \quad Recall} \tag{6}$$

Beside these, there are two types of curves specifically use in classification contexts and determine the classification of true positive classes and false positive classes, and these are ROC (Receiver Operating Characteristic) curve and PR (Precision–Recall) curve. In ROC curve true positive rate(TPR) is on y-axis and false positive rate is on x-axis. TPR is also equivalent to recall measure, whereas false positive is calculated by:

$$FPR = 1 - \frac{FP}{FP + TN} \tag{7}$$

And PR curve is simply a Precision Recall curve that plot the Precision on y-axis and Recall on x-axis.

Statistical significance test

To have statistical significance and validity of our proposed model classifiers and compare it with a baseline models to observe any significant difference in performance, we ran a 5 × 2cv paired t-test on our dataset. Although there are many statistical tests, but we applied this because it is a paired test, and in supervised classification learning, this means that the data of the proposed and the baselines approaches is the same, and it suits our context in which we applied same datasets separately on approaches. We use same disaster and COVID-19 datasets for the baseline and proposed model. As its name implies this test typically split the dataset into two parts (training and testing) and repeat the splitting(50% training and 50% testing) 5 times, in each iteration (Dietterich, 1998). In each of the 5 iterations, we fit A and B to the training split and evaluate their performance ($pAandpB$) on the test split. After this, it again rotates the test and train sets and computes performance again, which results in 2 performance difference measures:

$$p^1 = p_A^1 - p_A^1 \tag{8}$$

$$p^2 = p_A^2 - p_A^2 \tag{9}$$

Then it estimates the mean and variance of differences through following equations: mean is:

$$\overline{p} = \frac{p^1 + p^2}{2} \tag{10}$$

and variance is:

$$s^2 = (p^1 - \overline{p}^2)^2 + (p^2 - \overline{p}^2)^2 \tag{11}$$

The formula of computing t-test statistics for this test is as follows:

$$t = \frac{p1^1}{\sqrt{1/5\sum_{i=1}^{5} S_i^2}} \tag{12}$$

where $p1^1$ is p1 from very first iteration. The t-statistics assuming that it approximately follows as t-distribution with 5 degrees of freedom, and our hypotheses statements and threshold values are:

H0= Both the classifiers have same performance on this dataset.

H1= Both classifiers does not have same performance on this dataset.

Our threshold significance level for p-value is α =**0.05** for rejecting the null hypothesis that both classifiers have same performance on this dataset. Under the null hypotheses, t-statistics value approximately follows a t-distribution with 5 degrees of freedom, so its value should remain in a given confidence interval which is **2.571** for **5%** threshold, and it indicates that both classifiers have equal performance. If t-statistics value greater than this limit value, we can reject the null hypotheses and can say that performance of both classifiers are significantly different. In supervised learning, you can implement 5 by 2 fold CV paired t-test from scratch, but there is a package called MLxtend that implements this test and gives you t-values and p-values of two compared classifiers,[3] therefore, in its parameters we just gave the framework classifiers names and scoring mode which was mean accuracy.

4.4. Experimental setup

The experimental environment is windows 10, python 3.9, one specifically allocated high computing server with two graphic cards of NVIDIA Tesla T4 and it has 3 terabyte(TB) memory and 128 GB random access memory (RAM), which is suitable for processing large datasets with minimal time. We have experimented two types of datasets after pre-processing, and the hyper-parameters setting is given in training classifiers Section 4.2.3, 4.2.1 and 4.2.2. Based on parameters described in Bert embeddings Section 4.1, we ran our experiments on our GPU's and it takes 2 h at most for Bert embeddings, and less than an hour for classifiers learning.

4.4.1. Datasets

Selecting the accurate and appropriate datasets are important for achieving good performance for classification and evaluating our model. Therefore, experiments have been performed on two kinds of datasets: one is pandemic dataset which contains tweets about COVID-19 pandemic, another one is disasters related dataset which contains tweets about 7 disasters. COVID-19 pandemic related tweets collected from public repository[4] and this dataset mainly used for classification purposes to extract relevant information from COVID-19 tweets, there are 22 columns attached with this dataset, for our requirement we utilized 4 columns mentioned in Table 4. The another reason to choose this dataset is that it is large enough and unlabeled which is suited for our annotation layer in proposed framework.

Another dataset is disaster related dataset which is a collection of various disasters datasets collected from Imran et al. (2013), Olteanu et al. (2014). These tweets were collected during 7 crisis that occurred during 2012 and 2013 and he human-made crisis or natural disasters. These 7 crisis were 2012 Sandy hurricane, 2013 Alberta floods, 2013 Boston bombings, 2013 Oklahoma tornado, 2013 Queensland floods, 2013 Texas explosion and 2011 Joplin tornado. We need to evaluate our framework on different type of crisis, therefore we have utilized the accumulated dataset of these different disasters. The total tweets were 70k, with 6 crisis have binary categories of relatedness such as relevant, irrelevant, on topic or off topic except 2011 Joplin tornado tweets which has 3 classes. But for uniformity purpose with the pandemic dataset, we have labeled all 70k tweets into 5 different situational information classes to further analyze relatedness types. Statistical detail of datasets is given in Table 4.

5. Results analysis and discussion

In order to verify our proposed model in text classification task, we have examined Accuracy, Precision, Recall, and F1 scores of each classifier on disaster and COVID datasets. Also we determine the performance of classifiers with ROC and PR curves. First we analyze the results of our proposed model classifiers separately, then we compare the results of our proposed framework with other frameworks used in prior research studies.

[3] http://rasbt.github.io/mlxtend/user_guide/evaluate/paired_ttest_5x2cv/.

[4] https://www.kaggle.com/smid80/coronavirus-covid19-tweets.

Table 4
Dataset statistics in detail.

Dataset	Description
Disaster related tweets	Total 70k tweets with different categories of relatedness 1. Total 7 crisis related datasets each contains 10k tweets 2. Relevant (related to crisis), non-relevant (not related to crisis) 3. Tweets include tweet id, tweet content, time, tweet relatedness
Pandemic related COVID-19 tweets	Total 10 million tweets 1. Generated from March 1st to April 30th 2020. 2. Columns; created at, tweet id, text, source, 3. Keywords; coronavirus, coronavirusoutbreak, coronavirusPandemic, covid19

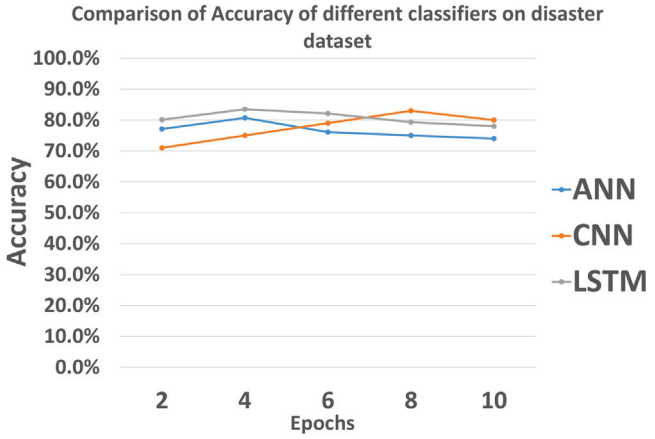


Fig. 4. Accuracy scores with different epochs values of all classifiers on disaster dataset.

5.1. Disaster dataset results analysis and discussion

The proposed framework were applied on disaster datasets, and accuracy results are shown in Fig. 4. In Fig. 4, it shows the accuracy rate of all classifiers on disaster dataset with respect to different epochs, the ANN classifier achieved highest 80.73% accuracy with 4 epochs, and when we see the confusion matrix values of ANN in Fig. 5(a), we can see that class 2 classification score is higher than others.

With CNN classifier on disaster dataset, we have achieved the highest accuracy score of 83.1% with the best fit of 8 epochs. Analyzing the accuracy ratio of CNN classifier in Fig. 4, it shows that after 8 epochs its accuracy rate decreases. The highest accuracy achieved with LSTM classifier is 83.51% on 4 epochs, after that it started decreasing as is shown in Fig. 4.

The confusion matrix of 5 classes with CNN classifier is shown in Fig. 5(b), particularly the class 1 achieved highest classification results, which indicates class 1 accurately classified the values with 88.2%. Initially CNN was used for classification of images, it has now enhanced to text classification contexts (Kim, 2014), as we have analyzed the accuracy rate goes up to a satisfactory level. Fig. 5(c) shows the confusion matrix values of all 5 classes by using the LSTM classifier on disaster dataset, it shows the highest classification rate of 87.74% by accurately classifying the text into class 1 categories, closely followed by class 2 accuracy rate of 85.51%.

The overall precision, recall and F1 measure scores are shown in Table 5, the comparison results among the classifiers indicate that CNN and LSTM achieved higher results than ANN classifier, because this disaster dataset contains data of different disasters, and it has more slang words used by users in the context of different situations, like floods have their own slangs, earthquake have their own, therefore, we can see with all classifiers, despite of this variation in data, the precision score is higher than the recall score. When you analyze the classifier separately, then LSTM and CNN scores are higher than the artificial neural network. You can see despite of LSTM having useful utility of keeping hierarchical representation of text for long term

Table 5
Evaluation measure scores of our proposed model with each classifier on disaster dataset, it shows the overall Precision, Recall, and F1 measure scores.

Classifiers	Precision	Recall	F1-score
T2L(ANN)	0.80	0.79	0.79
T2L(CNN)	0.84	0.82	0.83
T2L(LSTM)	0.83	0.82	0.82

duration, LSTM scores are slightly lower (1% lower in precision, 1% lower in F1) than the CNN, it indicates that in the disaster dataset, LSTM ability to keep long-distance dependence of texts does not matter much, because it contains data of 7 different types of disasters that changes with the time duration.

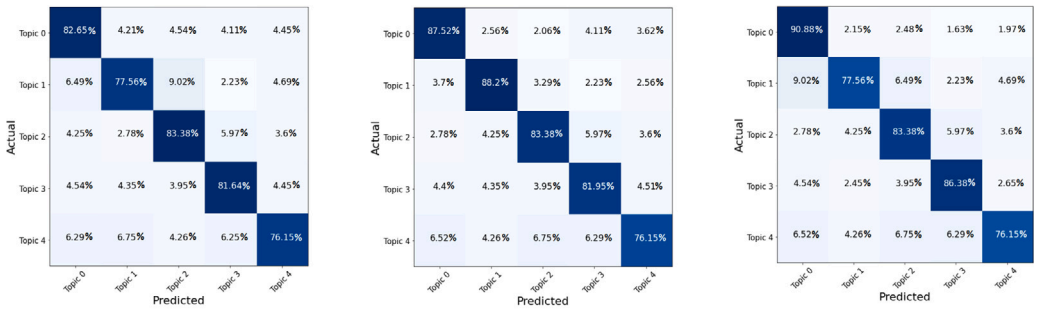
We also measured separability among classes by plotting ROC curve. In Fig. 6 it shows the ROC curve on disaster dataset by using highest accuracy classifier which was LSTM on disaster dataset. As our AUC threshold was 1 and in the ROC curve it shows that class 0 has highest AUC score which is 0.91 compared to rest of the classes which indicates that classifier has optimally classified text into class 0. Overall micro average is 82% and macro average is 86% which indicate that our classifier model is overall very good to distinguish classes among each other. Class 1 and class 4 AUC area is somehow close which is 0.76 and 0.75 respectively, and it also shown through their curves as these both are growing side by side. As these curves are not overlapping each other thus it indicates that they showing good performance in classification of categories.

Another measure to evaluate output of classification model is Precision–Recall curve, in classification, precision is a measure of only true class prediction and recall is a measure of how many true classes are classified from the whole document. Therefore, this PR curve demonstrates the tradeoff between Precision and Recall. Fig. 7 shows this tradeoff on disaster dataset of all classes with highest accuracy rate classifier which is LSTM. Results on disaster dataset indicates that overall micro average area is 0.94 shows the strong precision of classifying classes accurately. Class 0 has higher precision and recall area indicating the classifier success of true prediction with minimum error. Class 2 and class 3 has area score of 0.82 and 0.83 respectively and high precision and recall explains the classifier ability to predict classes in large numbers. Although class 1 and class 4 has lower scores than others but precision curve is high on certain threshold.

5.2. COVID-19 dataset results analysis and discussion

On COVID-19 dataset, we need to classify data into 5 classes, in Fig. 8, it shows that ANN classifier achieved accuracy score of 84.72% on 8 epochs, CNN best accuracy rate is 80.01% which is also with 8 epochs, and LSTM accuracy of 82.7% and on 6 epochs is somehow improved as compared to CNN.

As shown in Fig. 9(a) confusion matrix values of ANN, class 1 text is classified more accurately among all classes with a precision score of 92.25%, and class 0 has a lowest precision score of 61.59%. If we see the confusion matrix values of the CNN classifier in Fig. 9(b), class 1 has a higher precision score (92.25%) followed by class 4 (85.34%), it also



(a) ANN (b) CNN (c) LSTM

Fig. 5. Confusion matrix values five classes on **disaster dataset** by using classifiers: (a) shows the confusion matrix of ANN classifier, (b) shows the confusion matrix of CNN classifier and, (C) shows the confusion matrix of LSTM classifier.

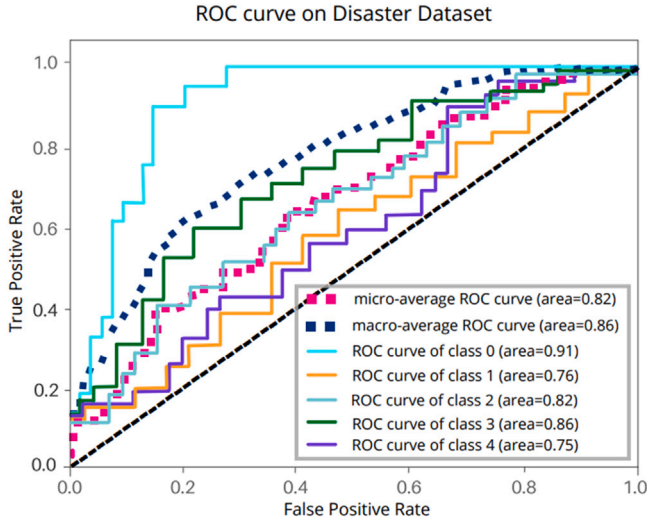


Fig. 6. Receiver Operating Characteristic graph for all classes of highest accuracy classifier on disaster dataset.

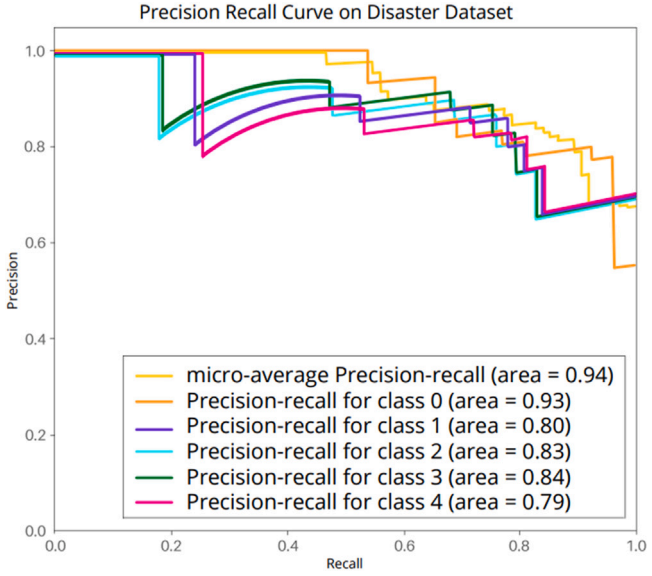


Fig. 7. Precision Recall curve graph for all classes of highest accuracy classifier on disaster dataset.

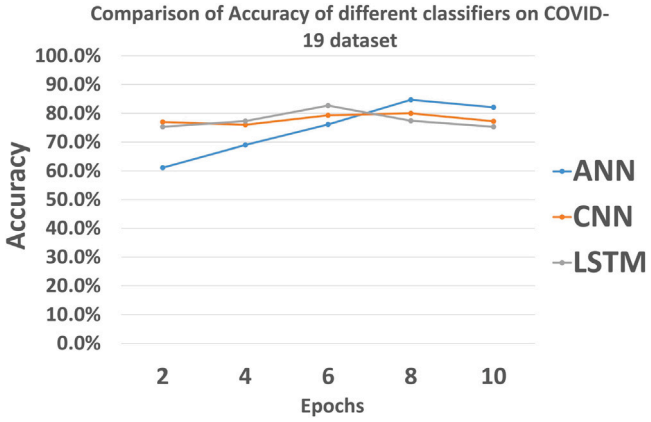


Fig. 8. Accuracy score with different epochs values of all classifiers on COVID-19 Dataset.

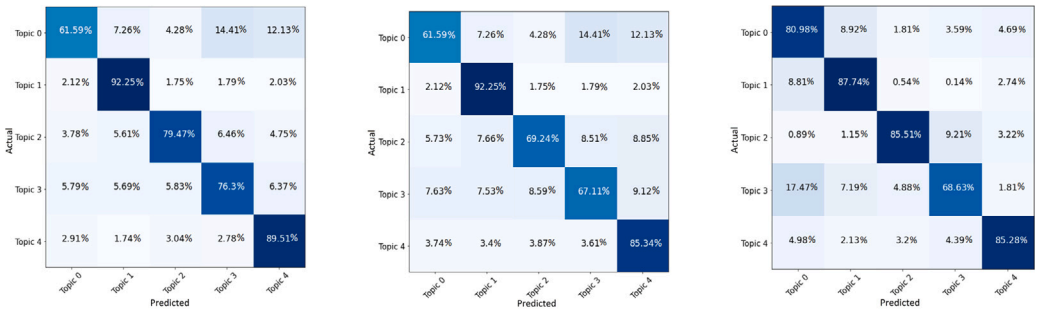
Table 6
Evaluation measure scores of our proposed model with each classifier on COVID-19 dataset, it shows the overall Precision, Recall, and F1 measure scores.

Classifiers	Precision	Recall	F1-score
T2L(ANN)	0.80	0.77	0.78
T2L(CNN)	0.75	0.72	0.73
T2L(LSTM)	0.82	0.76	0.77

shows the same trend in terms of class 1 precision score as compared with ANN classifier. In Fig. 9(c), when look at the confusion matrix values of LSTM classifier, it shows that Class 1 precision rate is higher than the other classes, class 3 has lower precision rate which is 68.83% as compared to precision of other classes. The classification error of class 3 is also higher with 17.47% of with class 3 text classified as class 1.

While comparing the overall scores of all classifiers on COVID-19 dataset shown in Table 6, ANN classifier achieved better accuracy than its other two counterpart's classifiers. LSTM accuracy score is slightly higher than the CNN, it indicates that on COVID-19 dataset long distance dependence of contextual information is reserved, which is one of the main benefits of LSTM algorithm, which results in yielding good accuracy rate which is 82.7% as compared to CNN classifier, which is 80.01%. Comparing the experimental results of ANN, CNN and LSTM classifier on COVID-19 dataset, it can be concluded that classification accuracy effect of ANN is better than CNN and LSTM, but the precision scores values that are shown in Table 6, in these values LSTM precision rate is higher than that of other two classifiers, which verified the LSTM ability to extract long distance dependency of text.

Similar to ROC graph on disaster dataset, we also measured the classification performance of classifier on COVID-19 dataset through ROC



(a) ANN (b) CNN (c) LSTM

Fig. 9. Confusion matrix values five classes on **COVID-19 dataset** by using classifiers: (a) shows the confusion matrix of ANN classifier, (b) shows the confusion matrix of CNN classifier and, (C) shows the confusion matrix of LSTM classifier.

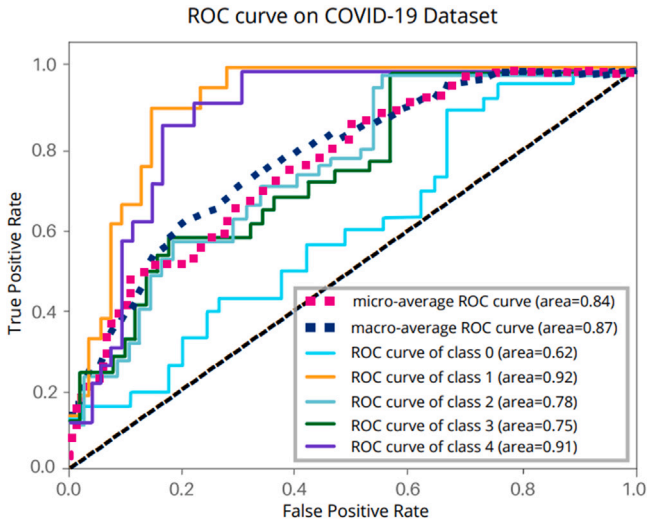


Fig. 10. Receiver operating characteristic graph for all classes of highest accuracy classifier on COVID-19 dataset.

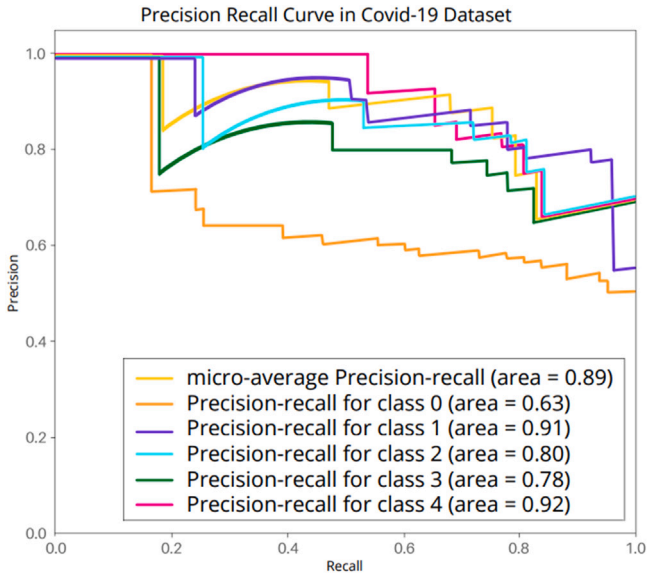


Fig. 11. Precision Recall curve graph for all classes of highest accuracy classifier on COVID-19 dataset.

graph. Fig. 10 shows all classes curves on COVID-19 dataset through highest accuracy classifier which is ANN. ON COVID-19 dataset class 1 and class 4 has highest AUC score with 0.92 and 0.91 respectively. It shows that model optimally classifies text into these two classes and performance is positively near to maximum threshold. It also indicates that classifier did a better job classifying class 1 as class 1 and class 4 as class 4. Overall when you see macro average score which is 0.87 which is the mean of individual classes Precision and Recall thus it indicates that model accurately classify text into all corresponding classes very well.

Fig. 11 shows the Precision–Recall tradeoff of all classes on COVID-19 pandemic dataset by highest accuracy rate classifier which is ANN. Class 4 which is related to cases and deaths has highest area score of 0.92. Although average area of 0.89 lower than the average area of disaster dataset but it is also satisfying score. Class 1 has PR area score of 0.91 and its curve has more variation that indicates this dataset is larger than the disaster thus its curve shows sudden rise in start and then after classifier learned outputs it shows stability to certain rate of precision. Class 0 has lowest area score of 0.63 and it curves indicating that it classify true prediction up to some point after that high recall means it still do true prediction but at the cost of negative prediction. Rest all other classes achieve satisfying results which shows the classifier ability to handle data which is large enough and of only one crisis.

6. Comparison with baselines approaches

The main disadvantage of mostly existing text classification methods is that they have human labeled dataset, and only some of them has automatic labeled dataset. Moreover, they only used internal feature extraction scheme of deep learning classifiers. In order to determine the effectiveness of our proposed T2L model proposed in this paper, we compare our model with following benchmark models:

1. Proposed by Xin et al. (2019), this method proposed an automated labeling approach by utilizing the number of named entities for text datasets, which can quickly retrieve a large number of text datasets in any field.
2. In Alrashdi and O'Keefe (2020), they proposed an automated approach using distant supervision to label large scale twitter dataset for crisis response. In distant supervision model they utilized initial keywords list and existing linguistic knowledge base FrameNet.
3. Si et al. (2018) proposed a model in which annotation is done through utilization of extended disaster lexicons. They created a master disaster lexicons by combining different datasets of same disaster type.
4. This paper Gupta and Joshi (2021) proposed an automatic labeling approach that make use of k-means clustering technique to

Table 7
Comparison of proposed framework with baseline approaches used in prior studies on disaster dataset, we have labeled the data by utilizing techniques proposed in baseline approaches.

Baselines	Accuracy	Precision	Recall	F1-score
Named entities recognition (Xin et al., 2019)	82.1%	80.9%	80.8%	81.4%
Distant supervision (Alrashdi & O'Keefe, 2020)	81.9%	82.7%	81.2%	79.4%
Disaster lexicons (Si et al., 2018)	80.1%	82.1%	79.5%	79.8%
Clustering approach (Gupta & Joshi, 2021)	79.2%	78.2%	78.3%	79.1%
Semantic knowledge (Chen et al., 2017)	77.2%	76.4%	75.3%	76.3%
LIWC lexicon (Karami et al., 2020)	76.3%	75.8%	76.1%	76.8%
Proposed				
T2L (ANN)	80.7%	80.1%	79.4%	79.1%
T2L (LSTM)	**83.5%**	83.1%	**82.0%**	82.2%
T2L (CNN)	83.1%	**84.0%**	**82.0%**	**83.1%**

create clusters based on underlying similar patterns, and label the tweets based on specific clusters.

5. Researchers in Chen et al. (2017) proposed an approach to automatically label training data for event extraction, by leveraging a semantic world knowledge (Freebase) and linguistic knowledge (FrameNet).
6. In this study, researchers Karami et al. (2020) identified the situational information during crisis, they label the data into different sentiments through LIWC.

We applied all these baseline approaches on our datasets, and for consistency purposes, we labeled the data into 5 classes, the same number of classes as in our proposed T2L framework. After applying the baseline approaches to label the data automatically, we compared these approaches classification results with our proposed T2L. The classification results of baselines approaches along with the proposed framework on disaster dataset are shown in Table 7.

A novel framework with an automated labeling approach and utilization of Bert embeddings with deep learning models for classification, is presented. It deals with sparse, user-oriented, short, and slang typed text in the context of different disasters. To reduce the human labeling cost and proposed a new automated way to label the data for better classification is the main objective of this framework. Therefore, in our framework we first proposed automatic annotation method, and then for classification, we utilized transformer-based Bert embeddings with ANN, CNN and LSTM deep learning-based classifiers. Evaluation measures results comparison with baseline approaches on our disaster dataset is shown in Table 7, from the results, it can be verified that T2L model proposed in this study performed well with LSTM and CNN yielding 83.5% and 83.1% accuracy respectively on disaster dataset, but one baseline approach (Xin et al., 2019) performed slightly better with 82.1% accuracy in comparison with our model with ANN classifier which yields 80.7% accuracy. Precision rate is higher than our T2L with ANN in couple of baseline studies, but still our proposed framework with CNN and LSTM comparatively better than all other baseline approaches is yielding 83.1% and 84.0% precision rate respectively, this is because while accurately classifying the results, labels should be most related to text and should be representative and contextual, and our proposed framework rightly depicted these properties during labels generation, while baseline approaches such as using LIWC lexicon (Karami et al., 2020) for labeling achieved only 75.8% precision and using linguistic knowledge (Chen et al., 2017) for labeling with 76.4% precision rate results are failed to achieve representative labels, resulting in poor classification performance as compared to our proposed approach. Specifically, these two approaches (Chen et al., 2017); (Karami et al., 2020) utilized only psychological lexicons, and word dictionaries that create labels related to only emotions, cognitive, and sentiment, ignoring the situational context of disasters, which considers to be the most important aspect during crisis (Wahid et al., 2021). Distant supervision and crisis keywords plays a vital role in annotation of texts and considers being one of the most successful techniques of labeling in NLP (Alrashdi & O'Keefe, 2020). Thus, we also took studies in our baseline approaches those utilized these techniques. The baseline approach by Alrashdi and O'Keefe (2020) utilizing distant supervision and by Si et al. (2018) leveraging lexicons related keywords for labeling the texts somehow increase the precision score yielding 82.7% and 82.1%, but still less than our proposed T2L precision rates of 83.1%, 84% with CNN and LSTM classifiers respectively, same goes with F1-scores which is also higher while applying our proposed framework.

By comparing the classification results of baseline approaches and the proposed framework on COVID-19 dataset, it can be concluded that proposed T2L model performs well and improves the classification results in terms of accuracy, precision, recall and F1 score. The accuracy scores are 84.2%, 82.7%, and 80.1% with ANN, LSTM and CNN classifier, respectively, as shown in Table 8. The precision rate of LSTM classifier achieved 82.7% highest score as compared to all other baseline approaches and the proposed method with ANN and CNN classifier, it is because LSTM uses the order information of neurons to express the hierarchical structure of sentences, the speed of model is faster, and it can fully extract the contextual information and long-distance dependency between texts.

Comparing the results of distant supervision technique proposed in Alrashdi and O'Keefe (2020) and named entity recognition (NER) labeling techniques proposed in Xin et al. (2019). It is investigated that on COVID-19 dataset labeling done through NER achieved good results 79.8% accuracy, 79.3% precision and 77.5% F1 score. As compared to annotation done through distant supervision technique which yields 78.4% accuracy, 76.2% precision and 77.4% F1 score, it is because in NER based method instead of labeling every word independently, it considers the correlation between labels in context and to accomplish this it implemented conditional random field to complete the annotating process. But instead of upper hand by the NER approach, our model yields higher results than these couple of enhanced labeling approaches, it is just because labeling based on the LDA topic model approach considers the context in the form of word probabilities, and shows probability distributions of every word in a topic, so dominant words with higher probabilities make the topic, and after dominant topic considers for labels in representing the specific sentence, our approach depicts proper labels that are contextually representative of texts. Lexicons based approaches are widely utilized by researchers to extract sentiments and keywords for labeling the texts, but the drawback of these approaches is that these only focus on sentiments and cognitive such as positive, negative, anger, emotions, and restricted to lexicons dictionary, by ignoring the situational information extraction that is necessary and meaningful during crisis response (Wahid et al., 2021). Thus, our proposed approach is comprehensive in a way that it deals with all aspects and it is contextual, because topics straight away extracted from the texts itself, and Bert embeddings yields the contextual representations while extraction of features, to keep the contextual and semantic meaning of texts. From the comparison results in Tables 7 and 8, we find out that proposed framework on disaster dataset, CNN precision rate is higher than LSTM, because disaster dataset is a collection of different disasters and LSTM ability to keep

Table 8
Comparison of proposed framework with baseline approaches used in prior studies on COVID-19 dataset, we have labeled the data by utilizing techniques proposed in baseline approaches.

Baseline	Accuracy	Precision	Recall	F1-score
Named entities recognition (Xin et al., 2019)	79.8%	79.3%	75.3%	77.5%
Distant supervision (Alrashdi & O'Keefe, 2020)	78.4%	76.2%	77.1%	77.4%
Disaster lexicons (Si et al., 2018)	78.2%	74.3%	76.2%	78.1%
Clustering approach (Gupta & Joshi, 2021)	76.1%	69.1%	74.3%	75.3%
Semantic knowledge (Chen et al., 2017)	70.6%	70.7%	67.2%	71.7%
LIWC lexicon (Karami et al., 2020)	73.5%	71.8%	71.7%	70.2%
Proposed				
T2L (ANN)	**84.2%**	80.2%	**77.2%**	**78.8%**
T2L (LSTM)	82.7%	**82.7%**	76.3%	77.2%
T2L (CNN)	80.1%	75.4%	72.1%	72.1%

Table 9
Comparison of each baseline model with the proposed model with highest accuracy classifier on same disaster dataset, and the *t*-value and *p*-value scores are listed.

	Proposed (T2L)	Proposed (T2L)
Baselines	***t*-statistics value**	***p*-value**
Named entities recognition (Xin et al., 2019)	3.134	0.0114
Distant supervision (Alrashdi & O'Keefe, 2020)	4.715	0.0034
Disaster lexicons (Si et al., 2018)	4.871	0.0046
Clustering approach (Gupta & Joshi, 2021)	5.156	0.0031
Semantic knowledge (Chen et al., 2017)	5.2110	0.003
LIWC lexicon (Karami et al., 2020)	5.3212	0.0021

Table 10
Comparison of each baseline model with the proposed model with highest accuracy classifier on same COVID-19 dataset, and the *t*-value and *p*-value scores are listed.

	Proposed (T2L)	Proposed (T2L)
Baselines	***t*-statistics value**	***p*-value**
Named entities recognition (Xin et al., 2019)	4.231	0.0043
Distant supervision (Alrashdi & O'Keefe, 2020)	4.134	0.0038
Disaster lexicons (Si et al., 2018)	5.651	0.0035
Clustering approach (Gupta & Joshi, 2021)	5.876	0.0024
Semantic knowledge (Chen et al., 2017)	6.321	0.0018
LIWC lexicon (Karami et al., 2020)	6.134	0.0011

long-distance dependency between texts does not matter much. Surprisingly on COVID-19 dataset, ANN achieved better accuracy score of 84.2%, which is better than the CNN and LSTM classifier, as COVID-19 dataset is of same pandemic. But, LSTM capability of keeping long-term dependency of texts shows effectiveness as the precision score of 82.7% is the highest among other proposed ANN, CNN and also higher than other applied baseline approaches.

6.1. Statistical test results and analysis

We have applied statistical significance test to investigate the comparison between proposed and baselines approaches and to see which framework has more statistical significance, we ran 5X2 CV paired statistical test on models. The configuration detail of paired statistical test with its parameters and hypotheses statements is given in . We compared baseline approaches with proposed framework (T2L) LSTM on disaster dataset, we compared LSTM classifier because mean accuracy of LSTM classifier is higher than the other proposed framework classifiers. The computed 5X2 CV paired *t*-test's *t*-value and *p*-values are shown in following Table 9.

When we conducted this statistics test on disaster dataset and compared the proposed framework classifier with the all baseline approaches with our threshold significance level, it is evident while analyzing the values in table that we can reject the null-hypothesis that every model comparing with the proposed model performs equally, as you can see *p*-values are less then the threshold values which is 0.05 therefore we have to accept the null hypothesis which is the performance of proposed and every baseline framework classifier is significantly different, since the *p*-value of every compared baseline is smaller than threshold.

Similarly to check the statistical significance on COVID-19 dataset, we ran this statistics test with same procedure and hyper-parameters on pandemic dataset. Since in the Table 8, it is evident that mean accuracy of ANN classifier is highest among the proposed framework classifiers, therefore we compared ANN classifier with all baselines while computing statistics test values. The *t*-values and *p*-values of statistics test are in following Table 9. On COVID-19 dataset too, while analyzing *p*-values it is evident that these are well less than the threshold, and if we see last 2 baselines approaches values in Table 10 these are very less than the threshold *p*-values, and others are also significantly less, and while analyzing the *t*-values we have seen these values are very well greater than the limit value of 2.571, thus it indicates that performance of baselines methods comparing with the proposed framework is significantly different and we can reject the null hypothesis that both comparing models have same performance. From the statistics test results we can assume that our model performance is significantly better than state of the art approaches that we have compared.

7. Conclusion and future work

This research work studies the problem of finding a new way to annotate the textual data to be used in classification tasks. Most existing methods used manual labeling or turned to unsupervised learning due to scarcity of labeled data. This proposed framework implements LDA topic model to extract meaningful and relevant topics, rank those topics through our newly developed ranking algorithm, then map those topics into labels to annotate the texts, and leverage transformer-based embeddings using Bert for feature extraction to improve the classification performance. Our method mainly consists of three parts named as annotation of texts through LDA topics, word embeddings of texts through Bert, classification of texts through deep learning classifiers. We evaluate our proposed method on two datasets of different type of disasters, datasets have noisy values, sparse data and slang language, and our proposed method handles that in an efficient way, thus improving the classification performance. From results it is evident that our proposed framework is ahead of baseline methods in classifying the textual data and improving the performance in terms of evaluation measures scores. When compared with other baseline approaches, results showed improvements while using T2L labels, with a highest 84.1% accuracy score and the highest F1 score of 83.1%. Therefore, we find that our proposed framework performs better than

the baseline approaches on accuracy, precision and F1 score, and it indicates that topic-oriented labels (T2L) extracted through LDA and T-TF-IDF ranking algorithm and feature extraction done through Bert embeddings can be leveraged as one of the valuable method to better classify the textual data. Our study is novelistic in a way that it leverages topic modeling technique to extract the topics, developed ranking algorithm to find out dominant topic from those topics, annotate tweets and did features extraction through Bert embeddings, thus paved a new way to create meaningful labels for texts and contextual embeddings and, improve the classification performance. This study purposefully targets the crisis-related data for crisis response in a way to classify relevant situational information, but this framework can be extended to other domains such as medical texts, education domains and citizen feedback texts. Although this LDA based topic model gives us very satisfying results, but in the future work, we will explore more topic modeling techniques to analyze topics to be used for annotation, and further increase the classification performance. Once tweets classified into situational information categories, we will also see in future work that how data oriented insights can be drawn from situational information types, such as determining the situational information classes scale and allocation of situational information types based on location area.

CRediT authorship contribution statement

Junaid Abdul Wahid: Conceptualization, Methodology, Software, Experiments, Writing – original draft, Writing – review & editing, Visualization. **Lei Shi:** Supervision, Writing – original draft, Writing – review & editing, Resources. **Yufei Gao:** Conceptualization, Methodology, Writing – review & editing, Investigation, Visualization. **Bei Yang:** Writing – review & editing, Supervision. **Lin Wei:** Writing – review & editing. **Yongcai Tao:** Writing – review & editing. **Shabir Hussain:** Writing – review & editing, Investigation. **Muhammad Ayoub:** Software, Experiments, Data curation. **Imam Yagoub:** Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by National Key R&D Program of China 2018 and Key Scientific and Technological Research Projects in Henan Province of China under grant number 192102310216. This work was also supported in part by the National Key Technologies R&D Program (2020YFB1712401, 2018YFB1701401), in part by the Nature Science Foundation of China (62006210), and in part by the major project of Zhengzhou Collaborative Innovation (20XTZX-009, 20XTZX-X010), the National Key R&D Program of China (2018******02) and the 2020 major public benefit in Henan Province (201300210500).

References

Alrashdi, R., & O'Keefe, S. (2020). Automatic labeling of tweets for crisis response using distant supervision. In *Companion proceedings of the web conference 2020* (pp. 418–425).

ALRashdi, R., & O'Keefe, S. (2020). Robust domain adaptation approach for tweet classification for crisis response. In M. Serrhini, C. Silva, & S. Aljahdali (Eds.), *Innovation in information systems and technologies to support learning research* (pp. 124–134). Cham: Springer International Publishing.

Athira, B., Jones, J., Idicula, S. M., Kulanthaivel, A., & Zhang, E. (2021a). Annotating and detecting topics in social media forum and modelling the annotation to derive directions-a case study. *Journal of Big Data*, *8*(1), 1–23.

Athira, B., Jones, J., Idicula, S. M., Kulanthaivel, A., & Zhang, E. (2021b). Annotating and detecting topics in social media forum and modelling the annotation to derive directions-a case study. http://dx.doi.org/10.21203/rs.3.rs-132773/v2.

Behera, R. K., Jena, M., Rath, S. K., & Misra, S. (2021). Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data. *Information Processing & Management*, *58*(1), Article 102435. http://dx.doi.org/10.1016/j.ipm.2020.102435, URL: https://www.sciencedirect.com/science/article/pii/S0306457320309286.

Bilbao-Jayo, A., & Almeida, A. (2018). Automatic political discourse analysis with multi-scale convolutional neural networks and contextual data. *International Journal of Distributed Sensor Networks*, *14*(11), Article 1550147718811827. http://dx.doi.org/10.1177/1550147718811827.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, *3*, 993–1022.

Caragea, C., Silvescu, A., & Tapia, A. (2016). Identifying informative messages in disaster events using convolutional neural networks. In P. Antunes, V. Banuls Silvera, J. Porto de Albuquerque, K. Moore, & A. Tapia (Eds.), *Proceedings of the International ISCRAM Conference*, *ISCRAM 2016 conference proceedings - 13th international conference on information systems for crisis response and management*. Information Systems for Crisis Response and Management, ISCRAM.

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems* (pp. 288–296).

Chatsiou, K. (2020). Text classification of manifestos and COVID-19 press briefings using BERT and convolutional neural networks. arXiv preprint arXiv:2010.10267.

Chen, Y., Liu, S., Zhang, X., Liu, K., & Zhao, J. (2017). Automatically labeled data generation for large scale event extraction. In *Proceedings of the 55th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 409–419). Vancouver, Canada: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/P17-1038, URL: https://aclanthology.org/P17-1038.

de Carvalho, V. D. H., Nepomuceno, T. C. C., & Costa, A. P. C. S. (2020). An automated corpus annotation experiment in Brazilian Portuguese for sentiment analysis in public security. In J. M. Moreno-Jiménez, I. Linden, F. Dargam, & U. Jayawickrama (Eds.), *Decision support systems X: Cognitive decision support systems and technologies* (pp. 99–111). Cham: Springer International Publishing.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*(7), 1895–1923. http://dx.doi.org/10.1162/089976698300017197.

Go, A., Bhayani, R., & Huang, L. (2009). *1*, *Twitter sentiment classification using distant supervision: CS224N project Report, vol. 1. no. 12*, (p. 2009). Stanford.

Greene, D., O'Callaghan, D., & Cunningham, P. (2014). How many topics? Stability analysis for topic models. In *Joint European conference on machine learning and knowledge discovery in databases* (pp. 498–513). Springer.

Gupta, I., & Joshi, N. (2021). Real-time twitter corpus labelling using automatic clustering approach. *International Journal of Computing and Digital Systems*, *10*, 519–532.

Imran, M., Castillo, C., Diaz, F., & Vieweg, S. (2015). Processing social media messages in mass emergency: A survey. *ACM Computing Surveys*, *47*(4), http://dx.doi.org/10.1145/2771588.

Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., & Meier, P. (2013). Practical extraction of disaster-relevant information from social media. In *Proceedings of the 22nd international conference on world wide web* (pp. 1021–1024).

Jelodar, H., Wang, Y., Orji, R., & Huang, S. (2020). Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach. *IEEE Journal of Biomedical and Health Informatics*, *24*(10), 2733–2742. http://dx.doi.org/10.1109/JBHI.2020.3001216.

Karami, A., Shah, V., Vaezi, R., & Bansal, A. (2020). Twitter speaks: A case of national disaster situational awareness. *Journal of Information Science*, *46*(3), 313–324. http://dx.doi.org/10.1177/0165551519828620.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 conference on empirical methods in natural language processing* (pp. 1746–1751). Doha, Qatar: Association for Computational Linguistics, http://dx.doi.org/10.3115/v1/D14-1181, URL: https://aclanthology.org/D14-1181.

Kim, S., Park, H., & Lee, J. (2020). Word2vec-based latent semantic analysis (W2V-LSA) for topic modeling: A study on blockchain technology trend analysis. *Expert Systems with Applications*, *152*, Article 113401.

Kralj Novak, P., Smailović, J., Sluban, B., & Mozetič, I. (2015). Sentiment of emojis. *PLoS One*, *10*(12), Article e0144296.

Krommyda, M., Rigos, A., Bouklas, K., & Amditis, A. (2021). An experimental analysis of data annotation methodologies for emotion detection in short text posted on social media. *Informatics*, *8*(1), http://dx.doi.org/10.3390/informatics8010019, URL: https://www.mdpi.com/2227-9709/8/1/19.

Li, H., Caragea, D., Caragea, C., & Herndon, N. (2018). Disaster response aided by tweet classification with a domain adaptation approach. *Journal of Contingencies and Crisis Management*, *26*(1), 16–27. http://dx.doi.org/10.1111/1468-5973.12194, URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/1468-5973.12194, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/1468-5973.12194.

Madichetty, S., & Sridevi, M. (2020). Classifying informative and non-informative tweets from the twitter by adapting image features during disaster. *Multimedia Tools and Applications*, *79*(39), 28901–28923.

Magdy, W., Sajjad, H., El-Ganainy, T., & Sebastiani, F. (2015). Distant supervision for tweet classification using YouTube labels. In *Proceedings of the ninth international conference on web and social media* (pp. 638–641). AAAI Press.

Menini, S., Aprosio, A. P., & Tonelli, S. (2021). Abuse is contextual, what about nlp? The role of context in abusive language annotation and detection. arXiv preprint arXiv:2103.14916.

Mohammed, S., Ghelani, N., & Lin, J. (2017). Distant supervision for topic classification of tweets in curated streams. arXiv preprint arXiv:1704.06726.

Moraes, R., Valiati, J. F., & Gavião Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications*, *40*(2), 621–633. http://dx.doi.org/10.1016/j.eswa.2012.07.059, URL: https://www.sciencedirect.com/science/article/pii/S0957417412009153.

Muhammad, P. F., Kusumaningrum, R., & Wibowo, A. (2021). Sentiment analysis using word2vec and long short-term memory (LSTM) for Indonesian hotel reviews. *Procedia Computer Science*, *179*, 728–735. http://dx.doi.org/10.1016/j.procs.2021.01.061, 5th International Conference on Computer Science and Computational Intelligence 2020, URL: https://www.sciencedirect.com/science/article/pii/S1877050921000752.

Mutanga, M. B., & Abayomi, A. (2020). Tweeting on COVID-19 pandemic in South Africa: LDA-based topic modelling approach. *African Journal of Science, Technology, Innovation and Development*, 1–10. http://dx.doi.org/10.1080/20421338.2020.1817262.

Naseem, U., Razzak, I., Musial, K., & Imran, M. (2020). Transformer based deep intelligent contextual embedding for Twitter sentiment analysis. *Future Generation Computer Systems*, *113*, 58–69. http://dx.doi.org/10.1016/j.future.2020.06.050, URL: https://www.sciencedirect.com/science/article/pii/S0167739X2030306X.

Nguyen, D., Al Mannai, K., Joty, S., Sajjad, H., Imran, M., & Mitra, P. (2017). Robust classification of crisis-related data on social networks using convolutional neural networks. In *Proceedings of the 11th international conference on web and social media* (pp. 632–635). AAAI Press, Publisher Copyright: © Copyright 2017, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.; 11th International Conference on Web and Social Media, ICWSM 2017 ; Conference date: 15-05-2017 Through 18-05-2017.

Olteanu, A., Castillo, C., Diaz, F., & Vieweg, S. (2014). Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the international AAAI conference on web and social media, vol. 8*.

Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and LDA topic models. *Expert Systems with Applications*, *80*, 83–93. http://dx.doi.org/10.1016/j.eswa.2017.03.020, URL: https://www.sciencedirect.com/science/article/pii/S0957417417301665.

Si, S., Win, M., & Aung, T. N. (2018). Automated text annotation for social media data during natural disasters. *Advances in Science, Technology and Engineering Systems Journal*, *3*(2), 119–127. http://dx.doi.org/10.25046/aj030214.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Vieweg, S. E. (2012). *Situational awareness in mass emergency: A behavioral and linguistic analysis of microblogged communications* (Ph.D. thesis), University of Colorado at Boulder.

Wahid, J. A., Shi, L., Gao, Y., Yang, B., Tao, Y., Wei, L., & Hussain, S. (2021). Identifying and characterizing the propagation scale of COVID-19 situational information on Twitter: A hybrid text analytic approach. *Applied Sciences*, *11*(14), http://dx.doi.org/10.3390/app11146526, URL: https://www.mdpi.com/2076-3417/11/14/6526.

Xin, Z., Tianbo, W., Haiqiang, C., Qiang, Y., & Xiaohai, H. (2019). Automatic annotation of text classification data set in specific field using named entity recognition. In *2019 IEEE 19th international conference on communication technology* (pp. 1403–1407). IEEE.