CRISIS RELATED SOCIAL MEDIA MULTICLASS TEXT CLASSIFICATION USING DNN

BRIJESH R NAMBIAR

A thesis submitted in fulfilment of the
requirements for the award of the degree of
MASTER OF SCIENCE IN MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

LIVERPOOL JOHN MOORES UNIVERSITY (LJMU)

NOVEMBER 2020

TABLE OF CONTENTS

DEDICATION

This work primarily contributes to the exploration and advancement of NLP and ML about multiclass text classification with DNN, pretrained contextual word embeddings, and its effectiveness in crisis management using social media short text corpus. Dedicating this work to all my fellow machine learning, NLP, and deep neural learning enthusiasts.

ACKNOWLEDGEMENTS

I would like to thank my thesis supervisor Ms. Harika Vuppala and Professor Dr. Manoj for their consistent support and guidance for the thesis report. I am also further thankful to Ugrad and Liver Pool John Moores University for all the considerate support, guidance, and providing this excellent opportunity to conduct and conclude this study as part of the Master of Science in Machine learning and Artificial Intelligence program.

ABSTRACT

Machine learning, natural language processing, and text classification models are now an integral and prominent platform to extract meaningful insights from social media short text corpus in the management of a crisis. These platforms with advanced ML and NLP capabilities can accurately identify events, situations, needs, significance, and other relevant information, as well as identifying non-relevant information about a crisis from social media short text corpus. These ML and NLP capabilities and its adaptation for crisis management have evolved considerably over the last decade with the latest entrant as deep neural networks and SOTA language models. Most of the previous and prominent studies and current platforms are on traditional models and the binary classification to understand as simple as relevant and non-relevant info.

In this study, the primary focus is deep neural network models with pretrained GloVe and BERT word embeddings. Deep neural networks with pretrained contextual word embeddings can provide significantly better results and accuracy for short text corpus with noisy data related to the crisis in multiclass text classification needs. There is a very high demand for more fast and efficient multiclass text classification models and a lot of ongoing studies showing highly favourable results.

In this study, we will be exploring the deep neural network models like CNN and BiLSTM with GloVe and BERT as word embeddings, with both class imbalance and balance date. We will observe, analyse, summarise the results, and discuss various models for multiclass tweet classification using CrisisNLP short text Twitter data corpus. The right model can identify accurate information, quick response time, and can enhance better decisions in a crisis.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANN	Artificial neural network
AUC	Area under the ROC curve
BERT	Bidirectional Encoder representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
DL	Deep learning
DNN	Deep neural networks
GloVe	Global Vector
GPU	Graphics processing unit
GRU	Gated recurrent unit
MIT	Massachusetts Institute of Technology
ML	Machine Learning
MLM	Masked Language Model
MLP	Multilayer perceptron
NLP	Natural Language Processing
OOV	Out-of-Vocabulary
RNN	Recurrent neural network
SOTA	State of the Art
SVM	Support Vector Machines

CHAPTER 1

INTRODUCTION

Machine learning and its natural language processing capabilities are now used extensively in crisis management by the government, other humanitarian rescue and aid agencies (Kejriwal and Zhou, 2020) to identify and extract crucial and critical time-sensitive relevant information. Social media has evolved as one of the prominent data sources for such an ML & NLP platform to extract meaningful insights is social media short text corpus (Madichetty and Sridevi, 2020). Social media starts to busts with data during any crisis. Twitter users tweet an average of 6000 tweets each second (Twitter usage statistics - internet live stats, 2020). At this point, social media (Uhl et al., 2017) seems to be one of the faster and potential media for data to be available quickly due to its usage nature from the consumers or concerned itself, rather than any planned medium like online sites or other media (Xavier and Souza, 2018). In earlier times, or almost a decade ago, studies and ML technology use were restricted to national security purposes and agencies, which was highly sophisticated, and complex compared to now. The current advancements in machine learning and NLP has made a significant shift and its adaption and use in crisis by all agents from affected to aid and rescue.

The biggest challenge for extracting insightful insights is the information is vast, coming in at a higher velocity, with a lot of noise and meaning that can get lost, which is critical and time-sensitive. The most important step to gather meaningful insights is to classify the data according to its urgency, priority, need, relevance, and then underlying intent. The classification need or problem is mainly of three types binary, multiclass, and multilabel. Categorizing information into relevant and non-relevant is a binary classification. To identify information according to its contextual meaning into different classes is multiclass text classification. Identifying the information class with the class relationship is a multilabel classification problem.

The latest advancements in NLP and Machine learning shows good promise and effectiveness in classifying a short text in conjunction with its temporal features, spatial features, contextual meaning, and can do not only binary also multiclass and multilabel text classification effectively. These NLP capabilities can go further ahead and predict potential future state or a possible state change, without losing its relevance. Currently, move advanced studies are happening in NLP space, its text classification capabilities, new NLP SOTA models are fast evolving, as well as the increased adaption of these capabilities to aid and rescue situations.

It's a well-known fact that short text social media corpus retrieved or consolidated from different sources, organized, prepared, and investigated using the latest deep learning technologies can present profound insights regarding the crisis as well as handling that crisis. The challenge for short text corpus from social medial is data sparsity or less relevant contextual data, more out of vocabulary words, and noisy data.

From the year 2018 onwards, ML and NLP space saw a sudden shift and explosion in SOTA language model space, new enhanced deep learning neural network models, and pretrained word embeddings available now producing quality benchmark results.

Type	Classification	Examples
Binary	Two types	Crisis / Not Crisis
Multiclass	More than two	Crisis / Not Crisis / Aid
Multilabel	More than two and sample can represent more than one label	Crisis / Not Crisis / Aid / Injury Crisis - Aid/ Not- Crisis - Aid / Injury – Crisis

Table 1.1 Examples of classification types

Table 1.1 describes about examples of different type of classification needs. Multiclass is about classifying more than one classes of information. Multilabel also would identify the dependency between the classes, so the information belongs to more class than one.

1.1 Background of the study

Text classification for crisis-related short texts from social media platforms like Twitter can be very cumbersome and time-consuming due to the raw nature and sparsity of the data. The challenges of text classification are sparsity of data, high dimensionality, word relationship, and semantics.

Feature extraction is one of the prominent areas where DL has several advantages over traditional ML. In traditional models an input with simple features, the learning steps will have complex features, and then another layer to map that features and to output. In deep learning, the features are learned at multiple levels. This reduces the need for feature engineering, to the number of features required, as this can be automatically learned, which itself is a huge advantage. Deep learning models on a high level comprises of two steps word vector representation or word embedding and then classification using a layered network.

The word embeddings which started its journey in 2013 by research at google aims to represent semantics in numerical form to perform further operations in it. The word apple can have different semantic meanings like the fruit, the logo, the company, the device. The world bank can have different meanings like the "bank of the river" is different from "going to the bank to do a withdrawal." This is the like most of the English words have, and more efficient word embeddings can preserve those semantic relationships. GloVe is an unsupervised word embedding. Both GloVe and BERT have the advantage of preserving these semantic relationships and positions comparing to traditional word embeddings. BERT uses attention-based transformers to do position encoding of the words.

The advantage of using a DNN is they do not have to be retrained, can generalize from the past trained data reducing the overall computational needs, can do online training, can train in mini-batches, and quickly adapt to new labelled data.

1.2 Problem Statement

The prominent studies and models are more focused on binary classification and using traditional word embeddings and models. There is a high demand or need for categorizing the information into its right class rather than just binary, and that the classification is accurate more than precise, and can be identified much faster or reduce delays, also with preserving its original intent. This study primarily focuses on the multiclass classification to ensure that the information without losing its relevance and intent can be identified in its right class. The objective is to observe and analyse improvements that can be made by the use of deep neural networks models like CNN and BiLTSM with GloVe and BERT as word embedding.

1.3 Aim and Objective

The objective of this study is to observe, analyse, and interpret between CNN and BiLSTM, as well as a hybrid CBiLSTM with contextual word embeddings as GloVe and BERT. These models can provide better accuracy in multiclass text classification by categorizing useful insights of relevance from social media Twitter short text corpus about the crisis.

- The first problem we are trying here to approach is to understand deep neural networks performance on the CrisisNLP short text twitter dataset and observe the results for multiclass text classification
- The second objective of this study is to evaluate the effectiveness of GloVe as word embedding on the most used deep neural network models on multitext classification
- The third objective is to understand the influence downsampling and oversampling on the results
- The fourth objective is to evaluate the effectiveness of BERT as word embedding on two most used deep neural network models on multitext classification
- The fifth objective is to evaluate the effectiveness of BERT as word embedding for a sequential combination of CNN and LSTM, a hybrid deep neural network

Finally, we will observe and will understand the results, and interpret how different models behave on various scenarios, the possible reasons for misclassified tweets, as these are crisis related tweets, recall is more or as important as precision

1.4 Scope of the study

There are different approaches for multitext classification in past, primarily on binary classification, sentiment analysis, and more on business-related problems and needs. This study focuses on using CNN, BiLSTM, and a combined CBiLSTM deep neural network models with GloVe and BERT as word embedding using short text twitter corpus from CrisisNLP dataset for crisis management needs.

The CrisisNLP dataset used in this study has human-labelled data from various crisis-related incidents like floods, hurricanes, and earthquakes in the English language. The scope includes combing the labelled dataset for the various crisis, exploring and pre-processing the data, building CNN, BiLSTM, and combined the CBiLSTM model with GloVe and BERT as word embedding. Train and evaluate the models on accuracy and interpret the results.

1.5 Significance of the study

Social media is now an integral part of our life and in a crisis its usage is at a peak as the urgency and importance to connect and exchange information need is high. The use of ML and NLP capabilities to gather intelligent insights from these social media short text corpus in several natural disasters for informational awareness, rapid decision and quick response can be seen in several recent events. Haiti 2010 Earthquake, Russian 2010 Wildfire, 2012 Hurricane Sandy of New York and Oklahoma for the 2013 Tornado are a couple of earlier examples. We can see that ML and NLP capabilities were extensively used in the management of recent pandemic and proven to be providing a great advantage in information management and decision making. Advancement in deep neural networks can effectively identify the classification of relevant information from a text corpus and improving the quality of information, avoid delays and errors in using that information to provide aid and rescue. More advanced capabilities and the use of new models and word embeddings can significantly improve the quality and results in such situations.

1.6 Structure of the study

This study includes a literature review and exploring research methodologies. Each major topic is organized as chapters in the report. In the second chapter, we explore background information and existing technologies for multiclass text classification. The literature review is conducted used systematic methods and information from the latest and reputed journals, articles, book with the new methodologies and evaluation metrics. The dataset used is CrisisNLP and its details are provided. The third chapter contains the research methodology and data pre-processing, models used, word embeddings, and evaluation metrics are explained and summarized. The fourth chapter contains the results of multiclass text classification using the above models and word embeddings. Finally, the results are summarized, concluded and future work explained.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

A systematic review of information available in the public domain for the last couple of years about crisis-related social media text classification using NLP, ML, DL would be explored during this study. The keywords like “social media crisis management”, “floods and social media”, “natural disasters and social media” were used for searching relevant information. The review includes books, journals, scientific articles to see the use of various ML and DL models being used for crisis-related text classification of popular social media platforms. These cover a broad technique, methodologies, theories, and practices used for the same.

These are text classification from binary, multilabel to multiclass scenarios. (Yu et al., 2019) states that social media data can become vast within a shorter period, and still the relevant data could be not much. This huge amount of data makes it almost impossible to do a manual intervention and would need text mining and NLP techniques to weed out irrelevant information and structure the data correctly for analysis and classification.

2.2 Existing approaches

Proposing deep neural networks to provide faster response, making better decisions, and handle the challenges of crisis (Nguyen et al., 2016) with less actual information. A CNN model with the usage of Twitter as a data source for binary and multiclass classification. The two key challenge is to ensure there is enough information available on the text as well as understanding what type of information is available or classifying the information. The proposal to use DNN for online learning than traditional methods is due to its advantage that it can learn model parameters as the new batches of labelled data arrive. To train the DNN model from prior labelled data and fine-tuned adaptively as newly labelled data arrived in new batches. CrisisNLP dataset was used for this classification study. Two embeddings were used in this study Crisis embeddings and Google embeddings. The epochs were 25, to avoid overfitting various dropouts of 0.0, 0.2, 0.4,0.5 and mini batches of 32,64,128 were used. The dataset is split into 70-10-20 for train, validation, and test batches. It was evident from this study that DNN has a definite advantage that every time a new batch comes, the model does not have to be retrained on the newly arrived data and avoiding the expensive computational task. Another advantage is feature engineering extraction without any manual intervention. CNN1 model with Crisis embedding was able to achieve an AUC score of 94.17.

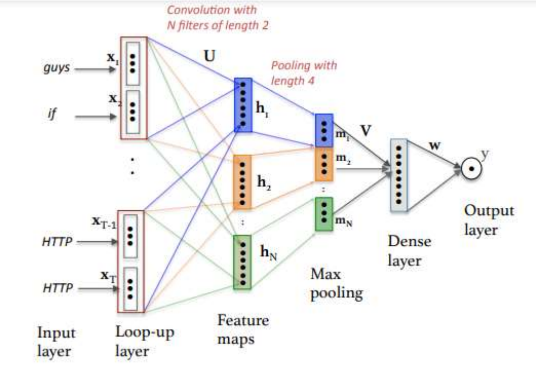
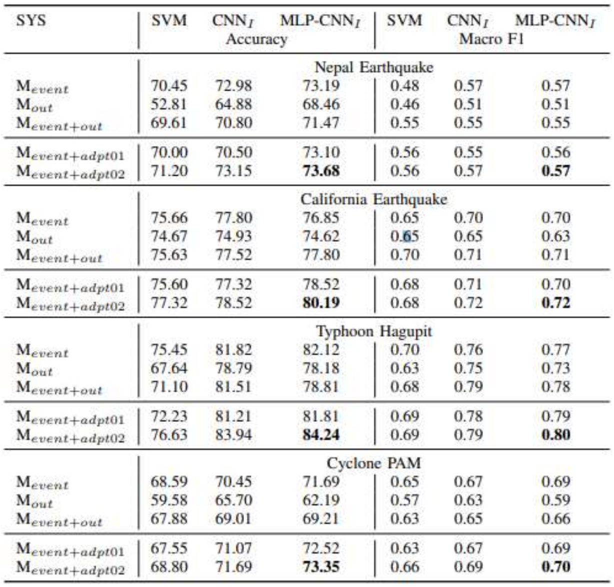


Figure 1: Convolutional neural network on a sample tweet: “guys if know any medical emergency around balaju area you can reach umesh HTTP doctor at HTTP HTTP”.

Figure 2.1 CNN model (Nguyen et al., 2016)

Figure 2.1 shows the convolutional neural network on a sample tweet. CNN models for binary classification to automatically identify methods and ways to identify disaster-related tweets and filter out non-relevant or non-informative ones (Caragea and Silvescu, 2016). The tweets are classified as informative and non-informative using CNN and conclude that using a deep neural network like CNN can outperform traditional models like SVM. A subset of Twitter data from six flood events CrisisLexT26 (Olteanu et al., n.d.) collection were used in this study. The results were compared with SVMs with features engineering with ANN and CNN model. The study indicated that CNNs were able to train from manually anointed tweets and were able to classify the “informative” and “non-informative” classification from train data with improved accuracy.

(Nguyen et al., 2016) proposes a CNN and MLP CNN model to automatically do multiclass classification from short text twitter corpus during a crisis. As the data is short, sparse, and noisy a distributed representation of words could result in improved generalization. Google word embedding and Crisis word embeddings were used for this study. The study tried to enhance the classification by experimenting with two domain adaption technique by weighting the out of event data or only using a subset of out of event data. The number of epochs were 25, experimented with 0.0, 0.2, 0.5 dropout rates and 32,64,128 mini-batch sizes. CNN's outperform the traditional classifiers in this study.

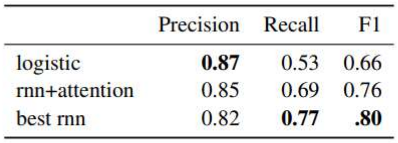


SYS	SVM	CNN$_I$	MLP-CNN$_I$	SVM	CNN$_I$	MLP-CNN$_I$
		Accuracy			Macro F1	
			Nepal Earthquake			
M$_{event}$	70.45	72.98	73.19	0.48	0.57	0.57
M$_{out}$	52.81	64.88	68.46	0.46	0.51	0.51
M$_{event+out}$	69.61	70.80	71.47	0.55	0.55	0.55
M$_{event+adpt01}$	70.00	70.50	73.10	0.56	0.55	0.56
M$_{event+adpt02}$	71.20	73.15	**73.68**	0.56	0.57	**0.57**
			California Earthquake			
M$_{event}$	75.66	77.80	76.85	0.65	0.70	0.70
M$_{out}$	74.67	74.93	74.62	0.65	0.65	0.63
M$_{event+out}$	75.63	77.52	77.80	0.70	0.71	0.71
M$_{event+adpt01}$	75.60	77.32	78.52	0.68	0.71	0.70
M$_{event+adpt02}$	77.32	78.52	**80.19**	0.68	0.72	**0.72**
			Typhoon Hagupit			
M$_{event}$	75.45	81.82	82.12	0.70	0.76	0.77
M$_{out}$	67.64	78.79	78.18	0.63	0.75	0.73
M$_{event+out}$	71.10	81.51	78.81	0.68	0.79	0.78
M$_{event+adpt01}$	72.23	81.21	81.81	0.69	0.78	0.79
M$_{event+adpt02}$	76.63	83.94	**84.24**	0.69	0.79	**0.80**
			Cyclone PAM			
M$_{event}$	68.59	70.45	71.69	0.65	0.67	0.69
M$_{out}$	59.58	65.70	62.19	0.57	0.63	0.59
M$_{event+out}$	67.88	69.01	69.21	0.63	0.65	0.66
M$_{event+adpt01}$	67.55	71.07	72.52	0.63	0.67	0.69
M$_{event+adpt02}$	68.80	71.69	**73.35**	0.66	0.69	**0.70**

Figure 2.2 Accuracy and F1 score (Nguyen et al., 2016)

Figure 2.2 shows the accuracy and F1 score from this study. The study concludes that time-critical analysis can be greatly helped with using a binary classifier to filter out noise first and then using it for humanitarian purposes. For humanitarian aid purposes then a multiclass text classification is proposed and CNN with domain adaption technique and use of neural networks shows highly favourable results.

Detection of crisis, as well as an explanation (Kshirsagar et al., 2017), is explored in this study. GloVe word embeddings 200D trained on Twitter data. The model used is a bidirectional GRU RNN, running model in each direction and concatenating the hidden states of each model to obtain a contextual word representation. An unconditional attention mechanism serves two purposes to act as a contextual document representation and score vectors to seed explanation. The dataset used is provided by chatbots on a variety of social media platforms and based on a clinical trial at MIT and provided through a research partnership. The crisis data contains suicidal and mental disorders and other relevant information.



	Precision	Recall	F1
logistic	**0.87**	0.53	0.66
rnn+attention	0.85	0.69	0.76
best rnn	0.82	**0.77**	**.80**

Figure 2. 3 Crisis detection performance (Kshirsagar et al., 2017)

Figure 2.3 shows that best RNN were able to get a 0.80 F1 score. The data was annotated by crowd workers. The text data was tokened using spacy, padded at 150 max length, used 200-dimensional embeddings, and 100 dimensional forward and backward GRUs. The learning rate was RMSprop of .001 and with a batch size of 256. A dropout with 0.1 is added to the final layer to avoid overfitting. 20 epochs were used for training, and validation data used is 10%. The results demonstrated neural networks achieving a 0.80 F1 score on crisis detections, with a single feed forward GRU layer.

“Deep Learning Approach towards Multi-label Classification of Crisis Related Tweets” (Aipe and Mukuntha, 2018) explores a deep CNN model for linguistic feature engineering and classification of tweets into different categories of crisis. This multiclass classification of labelled crisis tweets evaluated on benchmark datasets shows the state-of-the-art performance. CNN as a model is selected for its ability to handle distributed representation of words. This study also examines the effect of Twitter-specific linguistic knowledge on deep CNN. Pretrained 300 D word2vec model as word embedding is fed in a 250-feature convolutional layer with a filter size of [2,3,4]. To avoid overfitting dropout probabilities are applied to the final three layers of the model. Tweet dataset used was CrisisNLP. Along with word2vec word embeddings, Crisis NLP word embeddings and google news word embedding were used to feed the model. An OOV word dictionary is used to change it to its corresponding normalized words. 300 is chosen as hidden size and 37 is used as max length for padding and tokenization. The batch size used is 32, 3 to 6 epochs were used for experiments and a dropout rate of 0.5 is used for the final three layers of each component. The text semantics like hashtag and URL keywords or a combination of them were showing better results with deep CNN.

(ALRashdi and O’Keefe, 2019) discusses shifting from convolutional layers and to other neural networks with general and domain-specific word embeddings for multiclass text classification using short text corpus from the crisis for identifying the right information, the right decision, and faster response time. This study uses CNN and BiLTSM models with GloVe embeddings and Crisis embeddings.

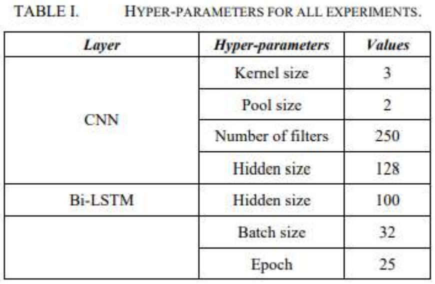


TABLE I. HYPER-PARAMETERS FOR ALL EXPERIMENTS.

Layer	*Hyper-parameters*	*Values*
CNN	Kernel size	3
	Pool size	2
	Number of filters	250
	Hidden size	128
Bi-LSTM	Hidden size	100
	Batch size	32
	Epoch	25

Figure 2.4 Hyperparameters (ALRashdi and O'Keefe, 2019)

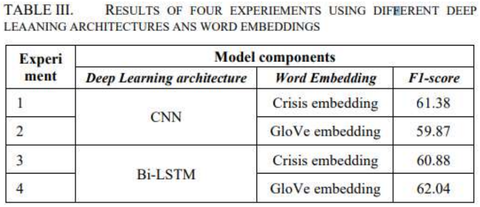


TABLE III. RESULTS OF FOUR EXPERIEMENTS USING DIFFERENT DEEP LEAANING ARCHITECTURES ANS WORD EMBEDDINGS

Experiment	**Model components**		
	Deep Learning architecture	***Word Embedding***	***F1-score***
1	CNN	Crisis embedding	61.38
2		GloVe embedding	59.87
3	Bi-LSTM	Crisis embedding	60.88
4		GloVe embedding	62.04

Figure 2.5 F1 score (ALRashdi and O'Keefe, 2019)

Figure 2.4 explains the hyperparameters used for this study and figure 2.5 indicates the results with CNN and BiLSTM with Glove and Crisis embedding, comparable performance, with BiLSTM with Glove having higher F1 score of 62.04.

2.3 Crisis related datasets

For this study, we are going to use human anointed twitter corpus data CrisisNLP (Imran et al., 2016). Social media platforms such as Twitter acts as a communication channel during emergencies like earthquakes, typhoons. The sudden response to a crisis itself people starts using Twitter for declaring the event and its state, and then on to the need for rescue, aid to be obtained or provided. The main task is human-labelled data. This dataset contains twitter from 19 different crises from 2013 to 2015. This dataset is the first largest word2vec word embeddings trained on 52 million crisis-related tweets. We will be using labelled data, combined from these multiple datasets, which totals around approximately 18000 plus rows when combined.

Note we have excluded pandemic related datasets, as they have slightly different labels and data. The datasets consist of floods, earthquake, cyclone, and hurricane from multiple countries from 2013 to 2015.

	Dataset Name	Row Count	% of Count
1	2015_Nepal_Earthquake_en_CF_labeled_data	3003	15.832762
2	2014_Philippines_Typhoon_Hagupit_en_CF_labeled_data	2010	10.597353
3	2015_Cyclone_Pam_en_CF_labeled_data	2004	10.565719
4	2014_Chile_Earthquake_en_CF_labeled_data	1932	10.186113
5	2013_Pakistan_eq_CF_labeled_data	1881	9.917225
6	2014_India_floods_CF_labeled_data	1820	9.595613
7	2014_Pakistan_floods_CF_labeled_data	1769	9.326725
8	2014_California_Earthquake_CF_labeled_data	1701	8.968208
9	2014_Chile_Earthquake_cl_labeled_data	1585	8.356619
10	2014_Hurricane_Odile_Mexico_en_CF_labeled_data	1262	6.653662

Table 2.1 Details of various datasets in CrisisNLP from major crisis

From table 2.1 we can see the folder structure of the data, number of rows of data in each dataset. The folder name also shows information about the year, event, and the country. The tweet information is manually labelled into 9 informative unique classes.

	Classes	Class description
1	Injured or dead people	Reports of casualties and/or injured people due to the crisis
2	Missing, trapped, or found people	Reports and/or questions about missing or found people
3	Displaced people and evacuations	People who have relocated due to the crisis, even for a short time (includes evacuations)
4	Infrastructure and utilities damage	Reports of damaged buildings, roads, bridges, or utilities/services interrupted or restored
5	Donation needs or offers or volunteering services	Reports of urgent needs or donations of shelter and/or supplies such as food, water, clothing, money, medical supplies or blood; and volunteering services
6	Caution and advice	Reports of warnings issued or lifted, guidance and tips
7	Sympathy and emotional support	Prayers, thoughts, and emotional support

8	Other useful information	Other useful information that helps understand the situation
9	Not related or irrelevant	Unrelated to the situation or irrelevant

Table 2.2 Multiclass labels and description

From table 2.2, we can see the various class information for the tweet texts, and description about each class.

2.4 Summary

The current approaches and prominent studies for multiclass text classification for identifying crisis-related information from social media short text corpus are explained here. Different crisis-related datasets, different models, word embeddings, and parameters were used for these studies with varying results. Most of the studies are based on classic models and vanilla deep neural network models with traditional word embeddings. The studies on multiple deep neural networks and a hybrid-based architecture, use of BERT as word embedding to preserve the semantics are never explored.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

This chapter describes approaches to do multiclass text classification for social media Twitter text corpus from a crisis. From recent past to earlier times, several successful studies regarding multiclass text classification have been published about traditional methods, feature engineering, and the use of vanilla deep learning techniques. This study would focus on using deep neural networks as well as a hybrid deep neural network model with GloVe and BERT word embeddings. The dataset used here is Crisis NLP and there are multiple studies done using this dataset. We have used evaluation metrics compared with baseline using this dataset and explaining the results of using the proposed hybrid model.

The multiclass classification pipeline is built on data exploration, text pre-processing, feature extraction, model selection, evaluation, finetuning, analysis, and interpretation of results. Multiclass problems need to ensure that only one label or category the text belongs to. The data exploration shows how various classes are balanced or distributed and class imbalance if any. Data is processed and cleaned, so that is suitable for feature extraction and tokenization preserving its originality. The processed data is tokenized and padded and fed into a baseline model and then into other models with GLOVE and BERT as word embeddings. The workflow is shown below.

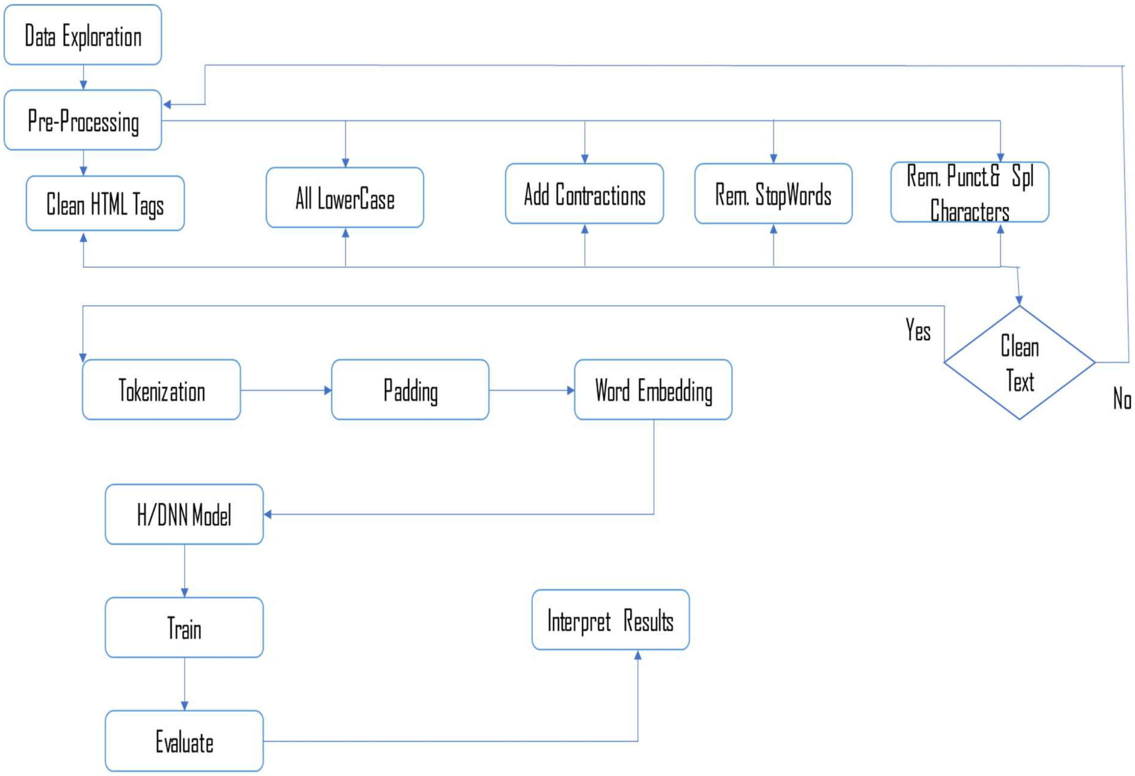


Figure 3.1 process flow

Figure 3.1 explains the overall process flow which will be followed for the study from processing and cleaning of the data, feeding to the model, and interpreting the results from various models.

3.2 Data Selection

The dataset being used is CrisisNLP (Imran et al., 2016) "Muhammad Imran, Prasenjit Mitra, Carlos Castillo: Twitter as a Lifeline: Human-annotated Twitter Corpora for NLP of Crisis-related Messages." The dataset is publicly available for download from the CrisisNLP website. The dataset consists of crisis-related tweets, and human-labelled tweets with OOV dictionary, and word2vec embeddings. The total data consists of more than 52 million tweets from 19 different crisis happened between 2013 to 2015 in various countries. These are collected as part of the AI for disaster response platform. The labels or categories used are a subset of annotations used by the United Nations for Coordination of Humanitarian Affairs. Volunteers were used to annotating messages and duplicates were removed.

For this study, we are going to use all the natural disaster labelled tweet text corpus and combine them into a single dataset, the two crises with diseases are excluded from this study as the labels are slightly different for them.

	Tweet id	Tweet text
0	'383600460340666369'	RT @Faiz_Baluch: #BalochistanEarthQuake Pakist...
1	'383790723222364161'	#Earthquake 2013-09-28 02:39:43 (M5.0) EAST OF...
2	'384232048124518400'	#earthquake M2.2: Puerto Rico region http://t....
18973	'593819547372638208'	RT @cnni: CNN obtains powerful images of baby ...
18974	'592137869482799104'	RT @ABCNews24: #NepalEarthquake update: The In...
18975	'591936980096659456'	#money #news #top #b Magnitude-7.9 quake hits ...
18976	'592344448887029761'	RT @PTI_News: Govt asks media houses to exerci...

Table 3.1 Tweet details

Table 3.1 shows the details of the combined data and some examples of information it contains. The combined labelled data from CrisisNLP dataset for various countries and events contain a total of 18977 rows.

3.3 Data pre-processing

Data needs to sanitize and processed with originality preserved before feeding into models. The following steps are followed

- Load the different labelled disaster files into a single combined dataset and randomize the combined dataset
- Drop the columns which are not relevant from the data frame, tweet_id and folder name of the dataset is not required for classification, and hence can be removed
- Evaluate for any null or missing values and process them if any. There are no missing or null values in this dataset
- UTF8 normalization, as characters can be represented in multiple ways, decode and encode the text back again as UTF8
- Lower case conversation as that can significantly improve the results of classification
- Remove HTML tags URLs (or Uniform Resource Locators) references to a location on the web, which mostly will not be relevant here for the crisis
- Screen name in Twitter, @userids are not of any particular use, remove them and replace them with userIds tag

- Remove special characters and punctuations as they do not add any specific relevance to text classification, and removing them or spacing them from words can provide a better and more accurate vector representation
- Contractions are shortened form of word or group of words and so its ideal to move it back to original form to get related and better vector representations, handle contractions using a contractions dictionary of common contractions
- Stop words are of no particular use and one of the common rules for better performance for classification models is to remove the stop words. Several tools and libraries are available for stop words removal. Here we will be using and downloading NLTK stop words and use that in the text tokens to remove stop words.

3.4 Word embeddings

Word embedding represents large size vectors with semantic importance. Global vectors (GloVe) (Pennington et al., 2014) and word2vec are the most popular and efficient word embeddings and widely used. In this study, we will GloVe and BERT as pretrained contextual word embeddings.

Encoded data is needed to feed to the model. (Wang et al., 2019) Word embeddings is a distributed representation of words or sentences in a vector form. Today many advanced embedding models can provide both semantic meaning and contextual meaning. If two words are similar in meaning, they should be near to each other, not apart.

GloVe is one of the most popular word embedding methods and uses co-occurrence statistics. They are already trained on around six billion tokens and available as pre-trained vectors, ready to use for text classification. Glove factorizes the co-occurrence probability matrix into two matrices, word ‘j’ appears in the context of the word ‘i’ is calculated for word pairs in text corpus ‘ij’. Glove seems to be consistently outperforming word2vec in various studies and results concluded. Glove license availability “Open Data Commons Public Domain Dedication and License (PDDL)” Glove is primarily trained on Gigaword and Wikipedia corpus, the pre-trained word vectors are of dimensions 50d, 100d, 200d and 300d vector.

BERT or Bidirectional Encoder representations from Transformers uses a bidirectional contextual word representation, and it was a breakthrough in text classification. BERT model aims to learn the context of a word not just its semantic meaning. The context would consider both the words before and after it. Here in this study, our interest here is only in pre-trained word embeddings. Bert needs to use special tokens. [CLS] first token of every sentence. [SEP] a sequence delimiter token. [MASK] Token for masked words in pre-training. Both tokens are needed for just one sentence. Note here BERT is not being used for classification and only word-embeddings and still, these special tokens are needed. Anon (2020) Input layer - vector of the sequence tokens along with special tokens are needed. Token embeddings are vocabulary IDs for each token. Sentence embeddings to distinguish between sentences. Transformer positional embeddings indicate the position of each word in sequence. (Alammar, 2020) study indicates to concatenate the last four layers for optimum results instead of 12 for optimal performance as word embeddings.

There are several models of BERT available, including tiny, mini, small, medium, base, and large. BERT-base would be used for this specific study. There are several other cases and multilingual models based on parameters and type of data trained. Each model consists of a specific set of layers, hidden states, heads, and parameters.

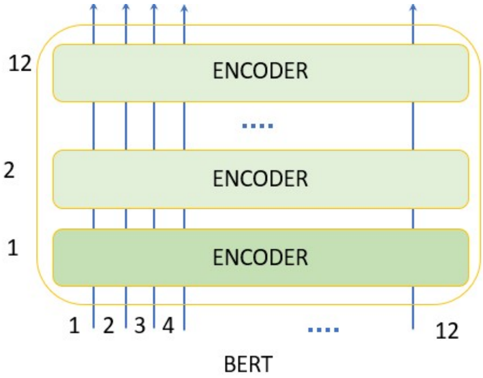


Figure 3.2 BERT Model

Figure 3.2 shows as BERT-Base model with 12 layers. BERT-Base Uncased model is selected for the following reasons,

- Tweet text corpus labelled data used from CrisisNLP dataset is on English language
- To limit the computational expense, the base model consists of 12 layers, 768 hidden states, 12 heads and 110m parameters
- BERT word embeddings are with GPU support

3.5 Modelling

Here we will design and use base DNN models and hybrid DNN models to do multi-class text classification, finetuning the parameters, training the data, evaluating the data, and summarizing the results. The split ratio between training and testing would be 80 – 20.

A convolutional neural network has a convolution layer, hidden layers, pooling layers, and fully connected layers. The convolution layer is the key aspect of this model, and through convolution features in the convolutional window are learned. Parameter sharing is applied to reduce the total number of parameters.

The pooling operation, average pooling, or max pooling is the next step after convolution to aggregate the spatial features. Max pooling can handle edges better, while average pooling can handle features more smoothly. For maxpooling the largest value in the convolution window is used, and for average pooling average of each window is used.

TextCNN (Kim, 2014) was able to extract excellent results with a simple CNN model, little hyperparameter tuning, and static vector wod2vec as embeddings. A simple one-layer CNN performed well, proving that unsupervised pre-training of word vectors can be an important element for overall efficiency.

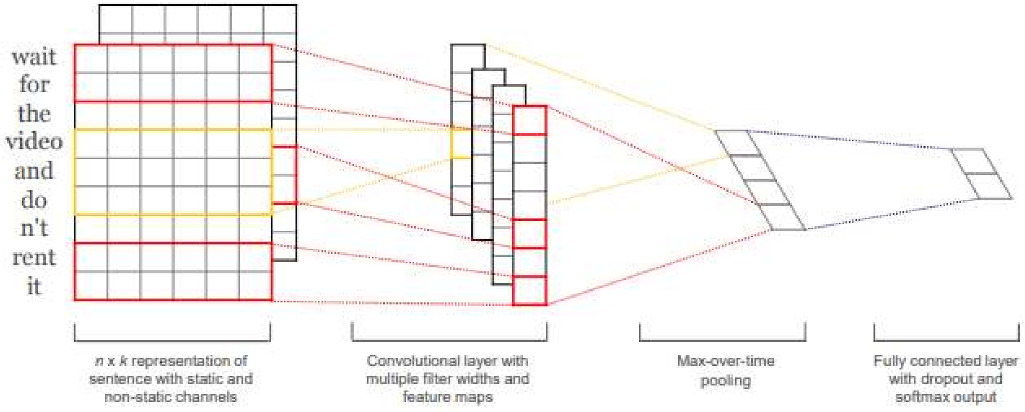

wait
for
the
video
and
do
n't
rent
it
n x k representation of sentence with static and non-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

Figure 3.3 KIM Text CNN Model

Figure 3.3 shows the Kim Text CNN model, it can handle distributed representation of words and learn the main features automatically at abstraction levels. The embedding matrix generated using pre-trained GloVe and BERT would be fed into a convolutional layer with a feature map of 32 and a filter size of [2,3,4]. The features are max pooled and concatenation and input into a fully connected dense layer with ReLU as the activation function. The architecture corresponds to the 9-class taxonomy to integrate with the features needed.

(ALRashdi and O'Keefe, 2019) Long Short-Term Memory networks (LSTM) can fix the gradient explosion, as well as can capture long term dependencies holding the contextual meaning with surrounding information. BiLTSM is one step ahead of LSTMs and can handle past and future directions. LSTMs can capture long-distance dependencies, and each unit consists of three gates for information to remember, forget, and pass to the next step. It can hold contextual semantics of each word and long dependencies between the words. BiLSTMs focus on the future and past directions on the input.

3.6 Summary

In this chapter, the various research approaches are detailed about multitext classification. The dataset selection, text pre-processing methods and various models with different word embeddings GloVe and BERT are discussed.

CHAPTER 4

ANALYSIS

4.1 Introduction

Here in this chapter, we will go deep dive into actual sanitization of text, exploration, visualization, and analysis of data, creating word embeddings extractions, and CNN, BiLSTM, and CBiLSTM model analysis.

4.2 Dataset description

The combined dataset we created from CrisisNLP dataset has approx. 18967 rows and two columns, the columns remaining in dataset of relevance are,

	Label	No of rows	% of Count
1	“other_useful_information”	5287	27.87%
2	“donation_needs_or_offers_or_volunteering_services”	3001	15.82%
3	“injured_or_dead_people”	2738	14.44%
4	“not_related_or_irrelevant”	2391	12.61%
5	“sympathy_and_emotional_support”	2076	10.95%
6	“infrastructure_and_utilities_damage”	1431	7.54%
7	“caution_and_advice”	1057	5.57%
8	“displaced_people_and_evacuations”	573	3.02%
9	“missing_trapped_or_found_people”	413	2.18%

Table 4.1 Label information in the dataset

Table 4.1 shows the distribution of classes in the dataset. The dataset consists of two columns,

1. **‘text**’ – tweet text from different users, different countries, different events
2. ‘**label’ –** 9 classes of information available in this label

The total combined rows in dataset is 18967 rows and 2 columns

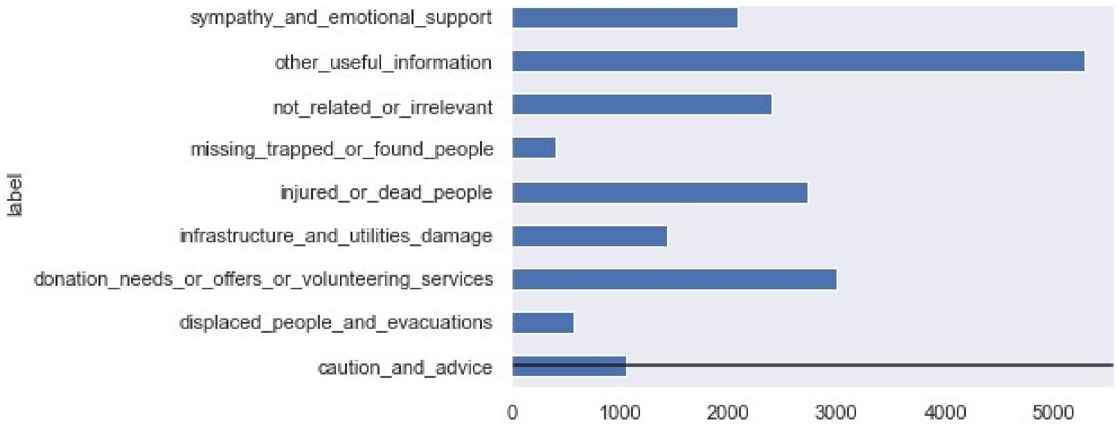


Figure 4.1 label class distribution

Figure 4.1 shows the class distribution of the data. From the analysis of the data is clearly imbalanced, with some of the classes with only 2 to 3% representation. In this study we will explore both original class imbalanced data and balanced data with down sampling and oversampling done.

4.3 Data Preparation

Data needs to be sanitized and prepared before tokenization, padding, and vectorization so that noise can be reduced without losing relevant information. This section will detail about various steps in data preparation.

4.3.1 Data Cleaning

The combined data from various events, first needs to be cleaned. We will be normalizing UTF8, removing punctuations and other special characters removed, converted to lower case, removing URLs, contractions changed into normal form, and stop words will be removed as they do not have contextual meaning. Example of texts after pre-processing.

	Text before cleaning	Text after cleaning
1	RT @JigarShahDC: #Solar lamps only public ligh...	rt usrId solar lamps public light nepal ; elec...
2	Another earthquake registered in Chile, magnit...	another earthquake registered chile magnitude ...
3	RT @WWECreative_ish: If Big Show &amp; @TheMar...	rt usrId big show amp ; usrId running cause ma...

4	Alaska Airlines halts flights from LA to Cabo ...	alaska airlines halts flights la cabo san luca...
5	News Update: Northern California Experiences M.	news update northern california experiences po..

Table 4.2 text data after cleaning

Table 4.2 shows the data before and after cleaning. The texts after cleaning looks much better meaningful data without losing its relevance and context.

4.3.2 Explore and visualize sanitized data

The next step is going through the data and grab the details of the text, sentences, and words, as this is a multiclass text classification need.

Word Length Range	No of sentence
0-5	205
5-10	2475
10-15	4850
15-20	7290
20-25	3734
25-30	403
30-40	10
>50	0

Table 4.3 sentence distribution

Table 4.3 shows the sentence distribution according it’s to length or number of words. The maximum length of a sentence is 40. There are 243773 words in total, with a vocabulary size of 22725 after the text cleaning step.

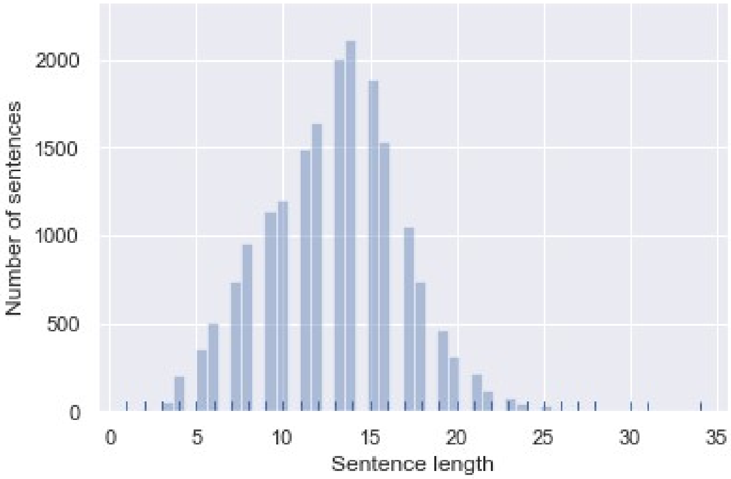


Figure 4.2 sentence distribution

Figure 4.2 plots the sentence distribution according to its length. Most data are within a length ranging between 10 to 20. For further adjustments of text, we will remove less frequent sentences with a word length less than 5.

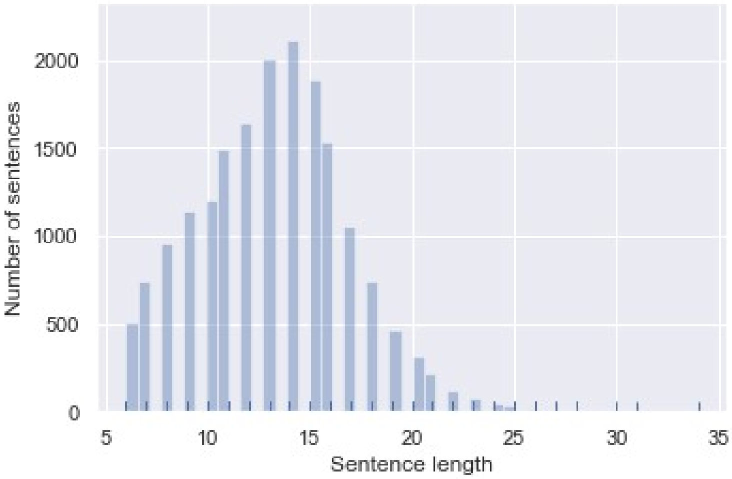


Figure 4.3 Sentence distribution length > 5 words

Figure 4.3 shows the sentence distribution after sentences with or equal to five words are removed. The text distribution looks much better after removing words with less than five words which is of no relevance as there is not enough information or meaning for those very small Tweets. From text, we will drill down more into words frequency.

Most frequent words	Count

Urls	24803
usrId	13155
Rt	8889
Earthquake	4194
DD	2867
Nepal	2647
Chile	2125
;	1846
!	1776

Table 4.4 most frequent words

Table 4.4 shows that the most frequent words seem to of places, or characters and not much of relevance or context.

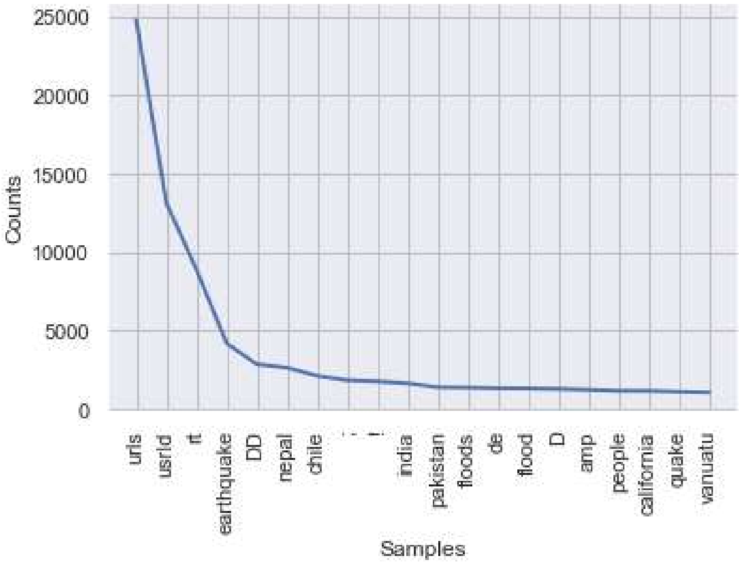


Figure 4.4 word frequency distribution

Figure 4.4 shows that most of the words is not having any meaning or without context for the classification need, we will do another adjustment to text to reduce the frequency of irrelevant words by removing them.

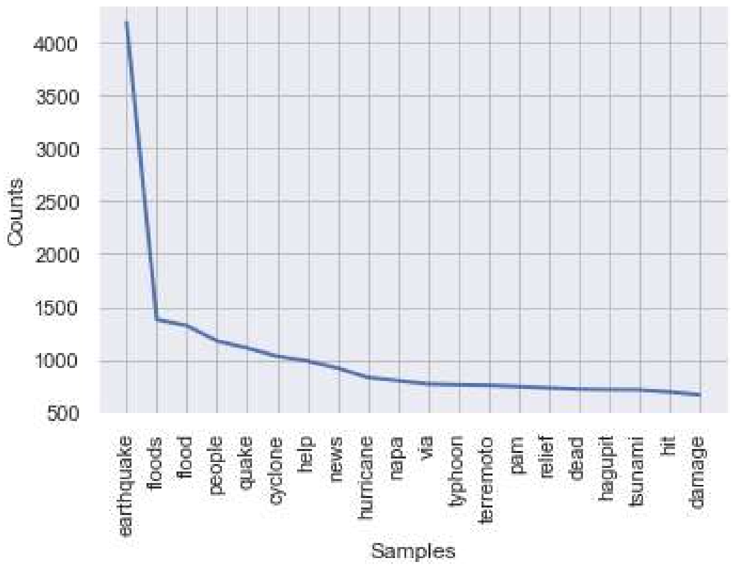


Figure 4.5 Word frequency distribution after further adjustments

Figure 4.5 shows after further manual clean-up of words and looks more cleaner with meaningful words high in frequency, not invalid characters or name or places. The final cleaned text data comprises of a frequency distribution of words with 22529 samples and a total of 165198 words without losing its original form and meaning.

4.4 Data transformation for analysis

The cleaned data is now split into test and train data and then converted into tokens, vectorized, and padded before feeding into models. For Glove embeddings, the hidden size will be 300 and for BERT it would be 768, which is by the standard for the BERT. Default Keras tokenizer and pad sequences would be used for Glove and Bert tokenizer and custom padding would be used for BERT. The max length for padding would be 40 so that all sentences are of equal length.

Data Type	Sample size	Percentage
Size of train data	14664	80%
Size of test data	3666	20%

Table 4.5 train and test data split

Table 4.5 show the test train split for the dataset and that would be at 80 – 20, after split train data contains of 14664 rows of text and labels, and test data contains 3666 rows.

4.5 Word embedding extraction

GloVe 300D “glove.840B.300d.txt” is downloaded and with the word index created, using python library NumPy the word vectors are extracted to word embedding matrix and will be used in the PyTorch nn.module embedding layer. Transformers Bert-base-uncased model is loaded as “bert = model_class.from_pretrained(pretrained weights) and will be used as embedding instead of the nn.module embedding layer and feed into models with a vector size of 768. As BERT is computationally expensive, we will be setting the default device to “cuda” to utilize GPU support and faster execution.

4.6 Class imbalance

As the data is highly imbalanced, we will also try to evaluate our various models with combination of GloVe and BERT using a modified balanced class dataset. The choice of size of the sample is assumed to be the size of the median class of the distribution of class at the middle. “sklearn.utils” library resample is used for this exercise (Dutta et al., 2018) and all labels are downsampled and oversampled to a size of 1919.

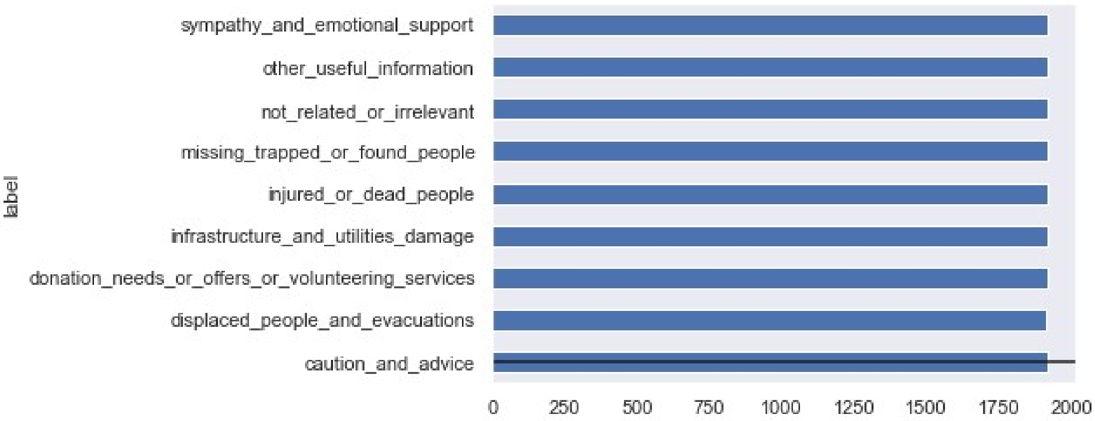


Figure 4.6 balanced class data

Figure 4.6 shows balanced class data after using resample. Using resample from “sklearn utils” all class data with higher dimensions were downsampled and with lower dimensions were oversampled to the median class size of 1919 and now represents a near perfect scenario where all class data are equal. This does not represent the general classification problem as always, the data would be imbalanced and this is for observing and studying the class recall, precision, accuracy and F1 score when data or class is imbalanced or balanced.

4.7 Model Analysis

The base model used for this study would be CNN and BiLSTM and then will evaluate its results with combined CBiLSTM model and interpret the results with the combination of GloVe and BERT as word embeddings, and also observe how various models behave or the difference on balanced and imbalanced class data.

	Parameters	Value
1	embed size	300 GloVe / 768 for BERT
2	max features	30000
3	max length	30
4	batch size	128
5	number of epochs	10
6	learning rate	Adam 0.001

Table 4.6 hyperparameters

Table 4.6 shows the hyperparameters used for this study. The hyperparameters were concluded using the values from prior studies what worked well as well as based on fine tuning and experimentation and its effect on the overall and class results.

	Model	Word Embeddings	Class data
1	CNN	GloVe	IMBALANCED
2	BiLSTM	GloVe	IMBALANCED
3	CBiLSTM	GloVe	IMBALANCED
4	CNN	BERT	IMBALANCED
5	BiLSTM	BERT	IMBALANCED
6	CBiLSTM	BERT	IMBALANCED
7	CNN	GloVe	BALANCED
8	BiLSTM	GloVe	BALANCED
9	CBiLSTM	GloVe	BALANCED
10	CNN	BERT	BALANCED

11	BiLSTM	BERT	BALANCED
12	CBiLSTM	BERT	BALANCED

Table 4.7 Model details

Table 4.7 shows the models, embeddings and balanced and non-balanced data class. There are 12 different scenarios or combinations used in this study using three different deep neural network model with two pretrained contextual word embeddings GloVe and BERT on actual class data and a perfectly balanced class data. The models and details as provided in the below tables.

CNN
CNN_Text(
(convs1): ModuleList(
(0): Conv2d(1, 32, kernel_size=(2, 768), stride=(1, 1))
(1): Conv2d(1, 32, kernel_size=(3, 768), stride=(1, 1))
(2): Conv2d(1, 32, kernel_size=(4, 768), stride=(1, 1))
)
(dropout): Dropout(p=0.2, inplace=False)
(fc1): Linear(in_features=96, out_features=9, bias=True)
)
Parameters: 222153

Table 4.8 CNN

Table 4.8 shows CNN, convolutional layer with a feature map of 32 and a filter size would be of [2,3,4]. The features are max pooled and concatenated and input into a fully connected dense layer with ReLU as the activation function. The output is with 9 layers as there are 9 classes for this classification need. For BERT the hidden size would be 768 as mentioned above, and for Glove it would change to 300.

BiLSTM
BiLSTM(
(lstm): LSTM(768, 64, batch_first=True, bidirectional=True)
(linear): Linear(in_features=256, out_features=64, bias=True)
(relu): ReLU()
(dropout): Dropout(p=0.2, inplace=False)
(out): Linear(in_features=64, out_features=9, bias=True)
)

Parameters: 444041

Table 4.9 BiLSTM

Table 4.9 explains BiLSTM, it has two LSTMs one taking in forward direction and other in backward direction, the hidden size is 768, dropout is 0.2 and output layer is 9.

CBiLSTM
CBiLSTM(
(convs1): ModuleList(
(0): Conv2d(1, 32, kernel_size=(2, 768), stride=(1, 1))
(1): Conv2d(1, 32, kernel_size=(3, 768), stride=(1, 1))
(2): Conv2d(1, 32, kernel_size=(4, 768), stride=(1, 1))
)
(lstm): LSTM(96, 64, batch_first=True, bidirectional=True)
(linear): Linear(in_features=256, out_features=64, bias=True)
(relu): ReLU()
(dropout): Dropout(p=0.1, inplace=False)
(out): Linear(in_features=64, out_features=9, bias=True)
)
Parameters 444041

Table 4.10 CBiLSTM

Table 4.10 shows the CBiLSTM, a combined model of CNN and BiLSTM sequentially with CNN having a hidden size of 768, and then feeding to BiLSTM with hidden layer of 96.

4.8 Summary

In this chapter, we explored the data and sanitized the data to be ready for vectorization. We also explored word embedding extraction, class imbalance handling, various hyperparameters, and models for this study.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Introduction

This chapter would train and evaluate the models using CNN, BiLSTM, and CBiLSTM models with GloVe and BERT as word embeddings on both balanced and imbalanced data using crisis-related short text Twitter corpus. The results obtained from these test and train model experiments are then used for observing, analyzing, and interpreting the performance of multiclass text classification models for crises.

5.2 Evaluation of Proposed Methods and Results

The results show comparative results between various models on GloVe, and with BERT. We will try to observe, analyze, and interpret various models in terms of accuracy, precision, and F1 score. Macro averages are more sensitive to imbalanced data, as it treats each class individually, and usually with lesser scores than micro, and will be used here to interpret the model performance.

We will try to generalize the different model's ability to do multiclass text classification with imbalanced data, how it handled classes with less representation, and how text-similarity of classes are being handled, the difference in results by using a highly balanced dataset.

Model	**Class balance**	**Accuracy**	**Precision**	**Recall**	**F1**
CNN + GloVe	IMB	0.72	0.72	0.63	0.66
BiLSTM + GloVe	IMB	0.71	0.70	0.63	0.65
CBiLSTM + GloVe	IMB	0.66	0.63	0.61	0.61
CNN + BERT	IMB	0.70	0.68	0.63	0.65
BiLSTM + BERT	IMB	0.67	0.65	0.58	0.60
CBiLSTM + BERT	IMB	0.68	0.65	0.60	0.61
CNN + GloVe	BAL	0.82	0.81	0.82	0.81
BiLSTM + GloVe	BAL	0.81	0.81	0.81	0.81
CBiLSTM + GloVe	BAL	0.80	0.80	0.80	0.80

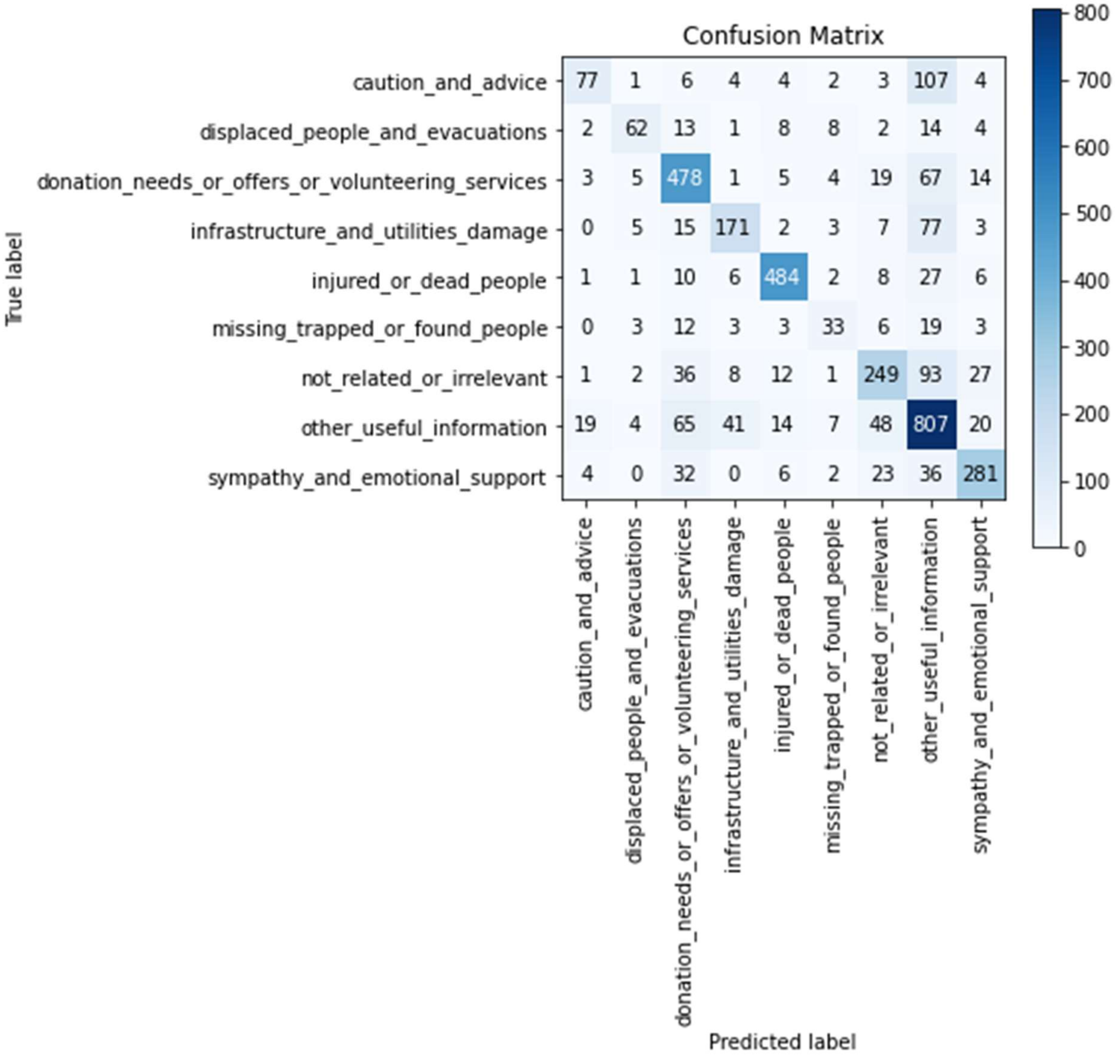
CNN + BERT	BAL	0.80	0.80	0.80	0.80
BiLSTM + BERT	BAL	0.77	0.77	0.77	0.77
CBLSTM + BERT	BAL	0.79	0.80	0.79	0.79

Table 5.1 results from experiments for various models

Table 5.1 shows the summary status of all the models and combinations. Overall GloVe as the word embedding performed slightly well than BERT in both imbalanced and balanced data, as BERT is a SOTA language model, this would be a future study topic to further experiment, observe, analyze and interpret that behavior.

In models, the CNN model seems to be slightly performing better comparing to BiLSTM or CBiLSTM on both balanced and imbalanced data, with an accuracy of 0.72, F1 score of 0.66 on imbalanced data, an accuracy of 0.82, F1 score of 0.81 on balanced data.

Let’s look at more details using the confusion matrix with the best-performed model – CNN with Glove on balanced and imbalanced data to understand class similarities and behavior on less represented classes.



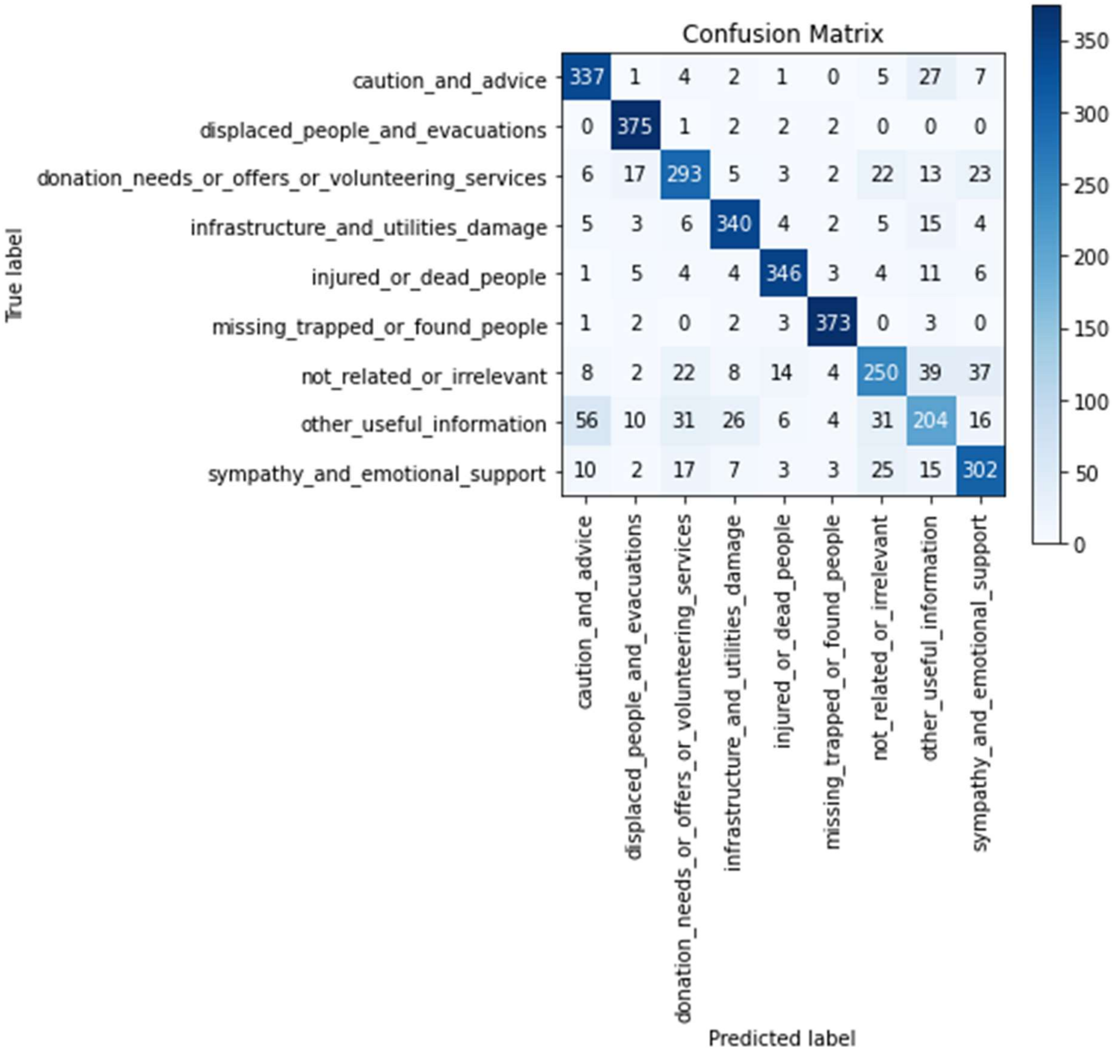
	caution_and_advice	displaced_people_and_evacuations	donation_needs_or_offers_or_volunteering_services	infrastructure_and_utilities_damage	injured_or_dead_people	missing_trapped_or_found_people	not_related_or_irrelevant	other_useful_information	sympathy_and_emotional_support
caution_and_advice	77	1	6	4	4	2	3	107	4
displaced_people_and_evacuations	2	62	13	1	8	8	2	14	4
donation_needs_or_offers_or_volunteering_services	3	5	478	1	5	4	19	67	14
infrastructure_and_utilities_damage	0	5	15	171	2	3	7	77	3
injured_or_dead_people	1	1	10	6	484	2	8	27	6
missing_trapped_or_found_people	0	3	12	3	3	33	6	19	3
not_related_or_irrelevant	1	2	36	8	12	1	249	93	27
other_useful_information	19	4	65	41	14	7	48	807	20
sympathy_and_emotional_support	4	0	32	0	6	2	23	36	281

Figure 5.1 Confusion matrix from GloVe CNN model with imbalanced data

Label name	precision	recall	f1-score	support
caution_and_advice	0.72	0.37	0.49	208
displaced_people_and_evacuations	0.75	0.54	0.63	114
donation_needs_or_offers_or_volunteering_services	0.72	0.8	0.76	596
infrastructure_and_utilities_damage	0.73	0.6	0.66	283
injured_or_dead_people	0.9	0.89	0.89	545
missing_trapped_or_found_people	0.53	0.4	0.46	82
not_related_or_irrelevant	0.68	0.58	0.63	429
other_useful_information	0.65	0.79	0.71	1025

sympathy_and_emotional_support	0.78	0.73	0.75	384
Accuracy			0.72	3666
macro avg	0.72	0.63	0.66	3666
weighted avg	0.72	0.72	0.72	3666

Table 5.2 classification metrics from Glove CNN model on imbalanced data



	caution_and_advice	displaced_people_and_evacuations	donation_needs_or_offers_or_volunteering_services	infrastructure_and_utilities_damage	injured_or_dead_people	missing_trapped_or_found_people	not_related_or_irrelevant	other_useful_information	sympathy_and_emotional_support
caution_and_advice	337	1	4	2	1	0	5	27	7
displaced_people_and_evacuations	0	375	1	2	2	2	0	0	0
donation_needs_or_offers_or_volunteering_services	6	17	293	5	3	2	22	13	23
infrastructure_and_utilities_damage	5	3	6	340	4	2	5	15	4
injured_or_dead_people	1	5	4	4	346	3	4	11	6
missing_trapped_or_found_people	1	2	0	2	3	373	0	3	0
not_related_or_irrelevant	8	2	22	8	14	4	250	39	37
other_useful_information	56	10	31	26	6	4	31	204	16
sympathy_and_emotional_support	10	2	17	7	3	3	25	15	302

Figure 5.2 Confusion matrix from Glove CNN on balanced data

	Precision	Recall	F1-score	Support
caution_and_advice	0.79	0.88	0.83	384

displaced_people_and_evacuations	0.9	0.98	0.94	382
donation_needs_or_offers_or_volunteering_services	0.78	0.76	0.77	384
infrastructure_and_utilities_damage	0.86	0.89	0.87	384
injured_or_dead_people	0.91	0.9	0.9	384
missing_trapped_or_found_people	0.95	0.97	0.96	384
not_related_or_irrelevant	0.73	0.65	0.69	384
other_useful_information	0.62	0.53	0.57	384
sympathy_and_emotional_support	0.76	0.79	0.78	384
Accuracy			0.82	3454
macro avg	0.81	0.82	0.81	3454
weighted avg	0.81	0.82	0.81	3454

Table 5.3 classification metrics from Glove CNN model on balanced data

From Figure 5.1 and 5.2, Table 5.2 and Table 5.3 its evident that CNN with Glove seems to be doing comparitively better from other models. Balancing the data can be seen as making a huge difference in improving the classification scores.

Some of the class data clearly shows similarity across the data, confusing the models, like “other_useful_information” with “caution_and_advice”, not_related_or_irrelevant with “other_useful_information” . and also as “other_useful_information” being the bigger class, this is very much a possible behaviour due to the similarity of data.

From generalized analysis, now we will look more closer into classes with very low representation “displaced_people_and_evacuations” (3%) and “missing_trapped_or_found_people” (2%).

Model	Precision	Recall	F1Score
CNN + GloVe	0.75	0.54	0.63
BiLSTM + GloVe	0.58	0.56	0.57
CBiLSTM + Glove	0.70	0.47	0.57
CNN + BERT	0.70	0.52	0.60

BiLSTM + BERT	0.64	0.41	0.50
CBiLSTM + BERT	0.55	0.42	0.48
CNN + GloVe + BAL	0.90	0.98	0.94
BiLSTM + GloVe + BAL	0.90	0.98	0.94
CBiLSTM + GloVe + BAL	0.91	0.99	0.95
CNN + BERT + BAL	0.92	0.99	0.95
BiLSTM + BERT + BAL	0.93	0.96	0.94
CBiLSTM + BERT + BAL	0.92	0.97	0.95

Table 5.4 displaced_people_and_evacuations

Model	Precision	Recall	F1Score
CNN + GloVe	0.53	0.40	0.46
BiLSTM + GloVe	0.63	0.33	0.43
CBiLSTM + Glove	0.34	0.34	0.34
CNN + BERT	0.59	0.43	0.50
BiLSTM + BERT	0.56	0.18	0.28
CBiLSTM + BERT	0.37	0.49	0.42
CNN + GloVe + BAL	0.95	0.97	0.96
BiLSTM + GloVe + BAL	0.96	0.96	0.96
CBiLSTM + GloVe + BAL	0.96	0.97	0.97
CNN + BERT + BAL	0.94	0.97	0.96
BiLSTM + BERT + BAL	0.93	0.96	0.94
CBiLSTM + BERT + BAL	0.91	0.97	0.94

Table 5.5 missing_trapped_or_found_people

From Table 5.4 and 5.5 its evident that CNN with Glove seems to be doing better also for classes less represented. Balancing the data can be seen as making a huge difference in improving the classification scores for low represented data.

5.3 Discussion

The dataset is with class similarity and class imbalance, we can see most of the errors are due to events being classified as not useful, so the challenge as always is the noise in dataset and the ways to handle that noise and still get critical and sensitive information without losing it.

The important need is recall than precision as its more important to understand a crisis need rather than loosing that important and critical info lost as irrelevant info than having couple of irrelevant info included as needed. Also some of the classes is highly similar and imbalanced, and also data contains a lot of noisy information, the challenge is to ensure critical and important data is not lost in the noisy data.

Introducing pre-trained BERT as word embeddings were not able to produce better results comparing to GloVe and would need further study, experimentation and analysis. This is also the case of introducing a combined CNN and BiLSTM model with no significantly better results comparing to CNN with GloVe, and would need further study, experimentation and analysis.

So far, we discussed about crisis related multiclassification needs, social media text as the potential data source, various deep neural networks with pretrained contextual word embeddings, experimentation and summarized it results. These efforts would add to the enhancement of multiclass classification techniques in crisis and use of further natural language processing and machine learning capabilities.

There is a lot room of improvement, like increasing the vocabulary size and training for a longer duration with more epochs may help in better scores. CrisisNLP dataset has its limitations, skewed, and need to explore other datasets in the future.

5.4 Summary

This chapter focussed on intrepreting and analyzing the results from various models with precision, recall and F1 scores. To further the reasons for the comaparitive performance of mutlliclassification with GloVe and BERT are justified. Highly imbalanced dataset with very less information and similair information is the possible reason for comparitive results and would need further experimenting, finetuning and experimentation. The challenge would always remain to handle the noise and similarity in datset and still getting precision without loosing recall or getting a perfect harmonic F1 score.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Introduction

In this chapter, we will discuss further about proposed multiclass classification problem, the models and a summary justification of metrics and how we can improve the same in the future.

6.2 Discussion and Conclusion

- CrisisNLP combined labelled dataset from various events and countries, and text pre-processing steps consist of data cleaning, and data transformation were completed before feeding into the various models.
- To handle class imbalance, the combine dataset was further oversized and downsized for higher and lower represented classes respectively.
- Pretrained contextual word embeddings GloVe and BERT were used, and BERT a SOTA language model were not able to produce significantly better results and need further study
- CNN and BiLSTM and a Combined CNN BiLSTM models were used for multiclass text classification in this research
- CBiLSTM a combined hybrid neural network failed to produce significantly better results and need further study
- In both balanced and imbalanced data, CNN with GloVe as the word embeddings were showing comparatively better results than the others
- The models provided a better performance on balanced data with much better accuracy, precision, and F1 Score.

6.3 Contribution to the knowledge

Most of the earlier studies are focused on traditional ML models and more on binary classification, about general situations and business needs rather than crisis as such. This study focused on deep neural architectures, multiclass text classification for crisis, and these are very recent advancements, compared to traditional and vanilla approaches. The use of BERT as pre-trained word embeddings for multiclass classification in crisis using short text corpus is entirely new in our best of knowledge. With the pre-trained word embeddings GloVe and BERT, it's seen that the training time for word vectors is significantly reduced. The results indicate that these models can handle multiclassification better than traditional models and word embeddings.

The use of BERT as word embedding and the use of hybrid deep neural networks are quite recent and highly advancing from semantic based approaches. In our knowledge and literature review, this is the first study using BERT as word embeddings for a crisis using CrisisNLP dataset using hybrid CNN and LSTM network for short text multiclass tweet classification.

6.4 Future Recommendations

This study has a lot of room for further improvement, and these are those areas we would like to explore further and improve our model capabilities on multitext classification for a crisis using short text corpus from social media.

- Other languages – Most of the studies are focused on English text, its ideal to explore crisis call for help and aid in other languages too
- Increasing the vocab size, adding other relevant crisis datasets available increasing the overall dataset and quality, handling OOV words. (Ray Chowdhury et al., 2020) (Alam et al., 2020) combing other crisis datasets and use the consolidated labeled data as a bigger corpus.
- Use of domain-specific crisis and tweet embeddings.
- Using other SOTA language models RoBERTa, ALBERT, DistilBERT (Kitaev et al., 2020) Handling class imbalance, synthetic oversampling, SMOTE Class sensitivity
- Using additional attention layers
- Using both CNN and BiLTSM as the encoder and concatenate the output - ConvBiLTSM

REFERENCES

Kejriwal, M. and Zhou, P., (2020) On detecting urgency in short crisis messages using minimal supervision and transfer learning. *Social network analysis and mining*, 101, p.58.

Madichetty, S. and Sridevi, (2020) Improved classification of crisis-related data on twitter using contextual representations. *Procedia computer science*, 167, pp.962–968.

Anon (2020) Twitter usage statistics - internet live stats. [online] Internetlivestats.com. Available at: https://www.internetlivestats.com/twitter-statistics/ [Accessed 4 Jul. 2020].

Uhl, A., Kolleck, N. and Schiebel, E., (2017) Twitter data analysis as contribution to strategic foresight-The case of the EU Research Project “Foresight and Modelling for European Health Policy and Regulations” (FRESHER). European journal of futures research, [online] 51. Available at: http://dx.doi.org/10.1007/s40309-016-0102-4.

Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K., (2018) *BERT: Pre-training of deep bidirectional Transformers for language understanding. arXiv [cs.CL].* Available at: http://arxiv.org/abs/1810.04805.

Kim, Y., (2014) Convolutional Neural Networks for Sentence Classification. arXiv [cs.CL]. Available at: http://arxiv.org/abs/1408.5882.

ALRashdi, R. and O’Keefe, S., (2019) Deep learning and word embeddings for tweet classification for crisis response. arXiv [cs.CL]. Available at: http://arxiv.org/abs/1903.11024.

Khatri, A., P, P. and M, D.A.K., (2020) *Sarcasm detection in tweets with BERT and GloVe embeddings*. *arXiv [cs.CL]*. Available at: http://arxiv.org/abs/2006.11512 [Accessed 24 Jun. 2020].

mlwhiz, (2020) *Multiclass Text Classification - Pytorch*. [online] Kaggle.com. Available at: https://www.kaggle.com/mlwhiz/multiclass-text-classification-pytorch?scriptVersionId=30273958 [Accessed 25 Nov. 2020].

Nguyen, D.T., Joty, S., Imran, M., Sajjad, H. and Mitra, P., (2016) *Applications of online deep learning for crisis response using social media information*. *arXiv [cs.CL]*. Available at: http://arxiv.org/abs/1610.01030

Nguyen, D.T., Mannai, K.A.A., Joty, S., Sajjad, H., Imran, M. and Mitra, P., (2016) *Rapid classification of crisis-related data on social networks using convolutional neural networks*. *arXiv [cs.CL]*. Available at: http://arxiv.org/abs/1608.03902.

Kshirsagar, R., Morris, R. and Bowman, S., (2017) Detecting and explaining crisis. arXiv [cs.CL]. Available at: http://arxiv.org/abs/1705.09585.

Aipe, A. and Mukuntha, N.S., (2018) *Deep learning approach towards multi-label classification of crisis related tweets*. [online] Iscram.org. Available at: http://idl.iscram.org/files/alanaipe/2018/2144_AlanAipe_etal2018.pdf [Accessed 28 Mar. 2020].

Ray Chowdhury, J., Caragea, C. and Caragea, D., (2020) Cross-lingual disaster-related multi-label tweet classification with manifold mixup. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*. Stroudsburg, PA, USA: Association for Computational Linguistics, pp.292–298.

Alam, F., Sajjad, H., Imran, M. and Ofli, F., (2020) Standardizing and benchmarking crisis-related social media datasets for humanitarian information processing. arXiv [cs.SI]. Available at: http://arxiv.org/abs/2004.06774.

Kitaev, N., Kaiser, Ł. and Levskaya, A., (2020) Reformer: The Efficient Transformer. arXiv [cs.LG]. Available at: http://arxiv.org/abs/2001.04451.

Xavier, C.C. and Souza, M., (2018) Extraction and classification of semantic data from twitter. In: *Proceedings of the 24th Brazilian Symposium on Multimedia and the Web - WebMedia '18*. New York, New York, USA: ACM Press.

Dutta, D., Paul, D. and Ghosh, P., (2018) Analysing feature importances for diabetes prediction using machine learning. In: *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, pp.924–928.

Minaee, S., (n.d.) Deep learning based text classification: A comprehensive review. [online] Available at: http://arxiv.org/abs/2004.03705v1

Liu, G. and Guo, J., (2019) Bidirectional LSTM with attention mechanism and convolutional layer for text classification. Neurocomputing, 337, pp.325–338.

Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Van Essen, B.C., Awwal, A.A.S. and Asari, V.K., (2020) The history began from AlexNet: A comprehensive survey on deep learning approaches. [online] Arxiv.org. Available at: https://arxiv.org/ftp/arxiv/papers/1803/1803.01164.pdf [Accessed 26 Aug. 2020].

Yu, M., Huang, Q., Qin, H., Scheele, C. and Yang, C., (2019) Deep learning for real-time social media text classification for situation awareness – using Hurricanes Sandy, Harvey, and Irma as case studies. International journal of digital earth, 1211, pp.1230–1247.

Caragea, C. and Silvescu, A., (2020) Identifying informative messages in disaster events using convolutional neural networks. [online] Cloudfront.net.

Olteanu, A., Vieweg, S. and Castillo, C., (n.d.) What to expect when the unexpected happens: Social media communications across crises. [online] Available at: http://dx.doi.org/10.1145/2675133.2675242.

Khan, F.H., Bashir, S. and Qamar, U., (2014) TOM: Twitter opinion mining framework using hybrid classification scheme. Decision support systems, 57, pp.245–257.

Burel, G., Saif, H., Fernandez, M. and Alani, H., (2017) On semantics and deep learning for event detection in crisis situations.

Imran, M., Mitra, P. and Castillo, C., (2016) Twitter as a lifeline: Human-annotated twitter corpora for NLP of crisis-related messages. [online] Mimran.me. Available at: https://mimran.me/papers/imran_prasenjit_carlos_lrec2016.pdf [Accessed 6 Sep. 2020].

Dhami, D., (2020) Understanding BERT — Word Embeddings - Dharti Dhami - Medium. [online] Medium. Available at: https://medium.com/@dhartidhami/understanding-bert-word-embeddings-7dc4d2ea54ca [Accessed 14 Sep. 2020].

Wang, Q., Liu, P., Zhu, Z., Yin, H., Zhang, Q. and Zhang, L., (2019) A text abstraction summary model based on BERT word embedding and reinforcement learning. Applied Sciences, 921, p.4701.

Anon (2020) BERT word embeddings tutorial · Chris McCormick. [online] Mccormickml.com. Available at: https://mccormickml.com/2019/05/14/BERT-word-embeddings-tutorial/ [Accessed 14 Sep. 2020].

APPENDIX A: RESEARCH PROPOSAL

Abstract

Anticipating and managing disasters is one of the fundamental reasons for the survival and prosperity of our race. Social media has attained tremendous popularity, widely adopted as a choice of interaction and communication medium, and a new entrant in this space. The advances in machine learning, deep learning, and new efficient transformer based NLP models are a game changer in identifying, managing, and preventing disruptive events. The identification or extraction of an event of significance is the most fundamental and crucial aspect of this process.

Table of Contents

1. Introduction

Event is an episode of interest with a diverse spectrum of importance. Events can be intended or disruptive ones. Classification of disruptive events and details around it is the first step in crisis or disaster management. The advent of social media, NLP, and deep learning presents an added platform and golden opportunity in event mining and analysis to decide for a precautionary and supportive response. Deep learning attention-based transformers and pre-trained models' platforms can infer and provide important intelligent insights about disruptive events from social media chatter.

Our world is affected by natural calamities from hurricanes to the tsunami over the centuries. We also emerged with techniques and systems to anticipate and control them with higher efficiency. Social media is the new participant and now acts as a powerful engine in broadcasting and receiving information in disruptive incident classification and analysis. Twitter users tweet an average of 6000 tweets each second (Twitter usage statistics - internet live stats, 2020) The amount of information available per second makes social media a huge source for potential intelligent insights. Advancement in NLP and deep learning is enabling social media as a platform for disaster identification and management. The most crucial step is distinguishing an event or situation of notable consequence.

Social media use and adoption is a fast-growing trend. (Yin et al., 2012) Situational recognition or identification of an event is one of the fundamental steps in disruptive event mining. Data retrieved or consolidated from multiple origins, organized, prepared, and investigated using the most advanced technologies can present profound insights regarding the condition as well as what to do about the condition. A condition can be extracted in conjunction with the temporal features, spatial features, meaning, and predict potential future progress. These were originally limited to defense and national security needs like identifying a contingency and possible sate change. That is an ancient story, is no longer restricted, and is currently utilized in a wide spectrum of disaster management.

2. Background and related research

The year 2012 (Zhou et al., 2020) witnessed tremendous improvements in NLP using deep learning techniques. The ImageNet and speech recognition with Switchboard moreover started the end of using statistical methods. As an instance, Bible from Microsoft demonstrated almost a human experience in rendering news from Chinese to English. R-NET and NL-NET of MSRA achieved human-quality results in SQuAD, EM score, and F1 score. Pretrained models like generative pre-training (GPT), bidirectional encoder representation (BERT), XLNET are showing meaningful advancement and delivering quality results in NLP.

The current deep learning RNN and CNN models for NLP are mostly based on the encoder-decoder. The attention-based pre-trained transformer model is the latest addition to NLP. Transformers can manage both long-range dependencies and sequence to sequence transactions. Bidirectional encoders are the ones moving the data forwards and backwards. The transformer are not directional and is a transduction model and it works on self-attention and pre-trained capability. The training is done by masking to identify those as an output. Masked language models the whole idea is to guess and enhance the prediction capabilities by masking a percentage of data at training itself. This is entirely different from the earlier sequence models. (Vaswani et al., 2020) More distinguished performance, shorter training time, higher quality, the capability to handle both huge or meagre training data are some of the benefits transformer models propose against traditional models. The Transformer matches the architecture utilizing heaped self-attention and point-wise, entirely connected layers for both the encoder and decoder. The transformer is not RRN rather attention mechanics. This prioritizes which steps are important Transformers utilizes main keywords as an extra input to the encoder (Hoang and Bihorac, 2020). Latest models use both prior and next token to consideration. The performance on small and ambiguous datasets shows huge promises in language modelling. These certainly is a huge leap from the prior word embeddings models.

The year 2019 (Sapé, 2020) witnessed tremendous growth in models and architecture variants for Transformer models. Some of them are Transformer-XL, GPT-2, Ernie, XLNet, RoBERTa, ALBERT, DistilBERT (Kitaev et al., 2020) and also another significant improvement is Reformer or the Compressive Transformer. This can significantly reduce the training time for longer sequences.

BERT is Bidirectional Encoder Representations from Transformers. (Devlin et al., 2018). BERT applies a “masked language model” or MLM to achieve a deep bidirectional transfer. XLNet exhibits an improvement above BERT as it seems not to ignore dependency among (Yang et al., 2019) masked regions and avoid pre-trained fine-tune disparity. XLNET applies Tranformer-XL autoregressive model toward pre-training.

3. Research Questions (If any)

XLNet exceeds BERT (Yang et al., 2019) in previous studies. The results were exemplary on some tasks like language inference, question and answering, sentiment analysis, and ranking. Can XLNet deliver more reliable outcomes than BERT on social media text extraction and interpretation for disruptive situations or events of significance irrespective of the scale? Is data sparsity, corrupted data, or large amount of data makes a difference in using these two models in text extraction for disruptive events?

4. Aim and Objectives

This study is to evaluate whether new state of the art transformer based language model like XLNET does have an advantage over BERT in text extraction from social media for disruptive and sensitive events, and address these challenges better, and quantify the results.

- How each model handle Data sparsity?
- How each model handles mumbled, mixed, and corrupted data?

5. Research Methodology

This work will be focussed on performance and quality of two prominent transformer-based model BERT and XLNET. Is there a significant advantage of using one over another in extracting events or situations of significance and contingency? Identify the effectiveness of the pretrained model on text classification and extraction provided the training data.

- Dataset identification, finding out a suitable set for BERT and XLNet
- Pre-processing text data for BERT, check missing and corrupt data, check for class imbalance, convert the raw text to numbers as these models cannot be input with raw text, adding tokens, padding, adding masks with the help of pre-trained tokenizer
- Build model for classification, context, and significance, create data-loaders for train, evaluation, and test, use default models or create models for classification, context, and significance, apply softmax for predicted probabilities of the output, select hyperparameters as recommended for fine-tuning, and train the model
- Evaluate the model using test dataset and capture the results, validate the accuracy, use confusion matrix for classification. Check predictions using some raw data
- Analyse results using various scoring mechanisms and conclude the results of the study

6. Expected Outcomes

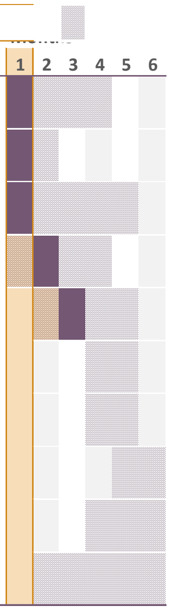
Quantitative analysis of how XLNet behaves better than BERT on social media event extraction for disruptive events

7. Requirements / resources

- Datasets (Zubiaga, 2018)
- Scientific journals and books
- Technical reports and articles
- Websites and web resources
- Mentoring support and guidance
- Online access to libraries
- GPU access

8. Research Plan

	PLANNED		*ACTUAL*		
TASK	START	DURATION	START	DURATION	% COMPLETE
Literaure Search	1	4	1	0	**25%**
Data Collection and Preparation	1	2	1	0	**25%**
Literature Review	1	5	1	0	**5%**
Investigate NLP transformer models	2	3	2	0	**5%**
Design State of Art NLP Models	3	3	3	0	**5%**
Train models	4	2	0	0	**0%**
Use State of Art NLP models	4	2	0	0	**0%**
Hyper parameter tuning	5	2	0	0	**0%**
Analyse & evaluate	4	3	0	0	**0%**
Complete report	2	5	0	0	**0%**



References

Anon (2020) Twitter usage statistics - internet live stats. [online] Internetlivestats.com. Available at: https://www.internetlivestats.com/twitter-statistics/ [Accessed 4 Jul. 2020].

Yin, J., Lampert, A., Cameron, M., Robinson, B. and Power, R., (2012) Using social media to enhance emergency situation awareness. IEEE intelligent systems, 276, pp.52–59.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. and Le, Q.V., (2019) XLNet: Generalized Autoregressive Pretraining for Language Understanding. arXiv [cs.CL]. Available at: http://arxiv.org/abs/1906.08237.

Zhou, M., Duan, N., Liu, S. and Shum, H.-Y., (2020) Progress in neural NLP: Modeling, learning, and reasoning. Engineering (Beijing, China), 63, pp.275–290.

Hoang, M. and Bihorac, O.A., (2020) Aspect-based sentiment analysis using BERT. [online] Aclweb.org. Available at: https://www.aclweb.org/anthology/W19-6120.pdf [Accessed 6 Jul. 2020].

Sapé, S.C., (2020) Beyond BERT? - towards data science. [online] Towards Data Science. Available at: https://towardsdatascience.com/beyond-bert-6f51a8bc5ce1 [Accessed 4 Jul. 2020].

Kitaev, N., Kaiser, Ł. and Levskaya, A., (2020) Reformer: The Efficient Transformer. arXiv [cs.LG]. Available at: http://arxiv.org/abs/2001.04451.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. and Le, Q.V., (2019) XLNet: Generalized Autoregressive Pretraining for Language Understanding. arXiv [cs.CL]. Available at: http://arxiv.org/abs/1906.08237.

(Sentiment analysis with BERT and transformers by hugging face using PyTorch and python, 2020)

Vaswani, A., Shazeer, N., Parmar, N. and Uszkoreit, J., (2020) Attention is all you need. [online] Arxiv.org. Available at: http://arxiv.org/abs/1706.03762v5 [Accessed 5 Jul. 2020].

Zubiaga, A., (2018a) A longitudinal assessment of the persistence of twitter datasets. Journal of the Association for Information Science and Technology, 698, pp.974–984.

Petrova, O., (2019) Understanding text with BERT. [online] Scaleway.com. Available at: https://blog.scaleway.com/2019/understanding-text-with-bert/ [Accessed 6 Jul. 2020].

Rizvi, M.S.Z., (2019) What is BERT | BERT For Text Classification. [online] Analyticsvidhya.com. Available at: https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/ [Accessed 6 Jul. 2020].