Journal of Theoretical and Applied Information Technology

15th December 2023. Vol.101. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LEARNING FROM DISASTER: DISASTER RELIEF MANAGEMENT USING DEEP LEARNING

NEKURI NAVEEN[1], DELTON M ANTONY[1], RAGHAVA M[2]

[1]school Of Computer And Information Sciences, University Of Hyderabad, Hyderabad, India.
[2]department Of Computer Science And Engineering, Cvr College Of Engineering, Hyderabad, India.

E-mail: naveencis@uohyd.ac.in, deltonmyalil@gmail.com, raghava.m@cvr.ac.in,

ABSTRACT

Our primary contributions in this article are the development of deep learning models for disaster management. The research work proposes three architectures with two variants each: A Long short-term Memory (LSTM) based model, a Convolutional Neural Network (CNN) based model, and a CNN-LSTM based model. Each of these architectures will have two variants built. The first variant is the one where the word embeddings are learned from the supervised data itself. The second one is where the word embeddings are pretrained. The results are presented based on empirical evidence and conclusions are highlighted.

Keywords: *Sentiment Analysis, CNN, LSTM, Multi-Model, Word Embeddings, Classification*

1. INTRODUCTION

The lack of timely and precise relief response in the event of a natural or man-made disaster is a major reason for the loss of life and property that follows the disaster. It can even be noted that the lack of proper response has caused more losses than in the event of the disaster itself. This lack of timely response is due to a lack of speed and agility in human thinking given a huge amount of information that bombards the human brain during such an unfortunate event. This is where the domain of Artificial Intelligence, particularly Deep Learning for Natural Language Processing can be employed to analyze the information which is generated during a disaster. This analysis can be used to deliver disaster relief to the needy and circulate the relevant information to the masses being affected by it. Here, we apply Deep Language Classification methods to Tweets generated during a disaster and classify the Tweets into categories like ‘caution and advice’ and ‘loss of life or property’ which can be used to provide aid. We start with Tweet preprocessing and then we create neural models from CrisisNLP datasets using CNN cells, RNN cells and its variants. , with and without pretrained word vectors and evaluate their performance. We found that CNN works best with tweet data to classify them into multiple labels. Finally, we evaluate all the models we make to find a high performing and accurate model which can be deployed to rapidly classify tweets as and when they appear when a disaster strikes, so that the disaster response teams can leverage the power of Deep Learning to save lives. Our primary contributions in this article are designing of six models - three architectures with two variants each: A Long Short Term Memory (LSTM) based model, a Convolutional Neural Network (CNN) based model and a CNN-LSTM based multi-model. Each of these architectures will have two variants built. The first variant is the one where the word embeddings are learned from the supervised data itself. The second one is where the word embeddings are pretrained.

1.1 Sources of Data:

Social Media: Since its invention, social networks have been a constant source of data related to any events big enough to attract attention. Out of all the social network platforms, microblogging websites have been a perfect source for short and to the point information which is ideal for exploitation by computing systems. This is due to the very nature of microblogs. As the name suggests, the data you get per microblog is short. But in this short set of words, people could express their sentiments, opinions, pass on news, knowledge and so on.

Tweets - the user generated short status messages from the microblogging site Twitter is one such example. Twitter users generate as many as 400 million tweets per day. During the course of an important event like a natural or manmade disaster, the number of tweets generated can spike even higher. User generated content on Twitter and other such microblogs contains information from rich and diverse topics. This data is considered to be real time data as Tweets appear as and when they are 'Tweeted' by the user. Mining such Tweets will give valuable insights to the people who are in desperate need of knowledge from large amounts of data which as a whole makes little sense. Human interpretation of data from Tweets is an impossible task when Tweets themselves come at a very high pace. Hence computer automated interpretation of Tweets will give the insights from the data to human in a much more comprehensible way, which can then be utilized by the people.

1.2 Twitter as a Lifeline

Disasters are events which are either natural or manmade that result in a large scale loss of life, property and disruption of services. Natural disasters like earthquakes could disrupt conventional communication means which require extensive infrastructure [1]. Reports on Hurricane Sandy states that the people relied on Social Media to communicate. Soon after Sandy struck, people used Twitter as a means to request and send for help, provide caution, offer volunteering work, request supplies and mobilize donations to help the unfortunate. Similarly, when Hurricane Harvey and Irma wrought havoc, Twitter again proved valuable in disaster management. Although this seemed promising at the time, the time and manpower required for any centralized body like the government to go through each of the Tweets and then sort them out based on priority or categorize them based on need is a difficult task when not automated. The amount of information coming in is just too large for humans to process on their own. We need a system which can identify the relevant information from a large number of irrelevant Tweets. Moreover, we need to categorize the relevant information into sub-categories which makes it easier to identify which Tweets are more important than others. The solution to this problem lies in the domain of Multilabel classification of text under Natural Language Processing (NLP). Twitter could act as a lifeline[2] during a crisis. Government based rescue - particularly those spearheaded by the military saves a lot of lives during a disaster. Furthermore, a well coordinated volunteering work done by the locals could be even more effective as they could already be at the affected area. For all of this to work systematically, there is a need for data - data which states what all is going on where, who all are trapped, where the hazardous zones are and so on. Twitter could be a genuine source of such data and language classification methods could categorize the information for us to prioritize and schedule the rescue.

2.3 Organization of the paper:

Section 2 presents a literature review related to the use of Twitter as a source of information during a crisis and the application of Deep Learning and other technologies on Tweets. Section 3 delves into the Proposed Model which explains the system architecture of the project and the methodology used in building the models. Section 4 presents the results obtained by this project and compares the models which were built. Section 5 offers conclusion and future works.

2. Literature Review

Most of the works focus on assessing the situation and some on rescue and planning. A rescue scheduling algorithm was proposed by authors in [16] on Hurricane Harvey. When Hurricane Harvey struck in the Houston area of Texas, United States, social media proved to be a notable communication platform for the people to ask for help and offer their services and volunteering. During the crisis, the emergency institutes were overloaded and could not support the large number of distress calls from the masses. In addition to it, they were not able to mobilize help and response systematically due to the inability to sort calls into different levels of priority. However, many people who needed help as well as the ones offering their services as volunteers put their contacts online on sites like Facebook and Twitter. [16] proposed three rescue scheduling algorithms that provides victims of disasters with quick response by mobilizing volunteering workers using these sites. The paper proposed First Come First Served policy, a priority queue based scheduling and a combined version of the two called Hybrid Scheduling. Hybrid Scheduling algorithm proved effective in reducing the average wait time in queue as well as the average time in the system for a request. It was successful in effectively connecting the hurricane victims to the volunteers.

ResQ [17] uses a Reinforcement Learning based algorithm to coordinate the volun- teers and victims during a disaster. resQ is a heuristic multi-agent

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

reinforcement learning scheduling algorithm which can efficiently schedule the rapid deployment of help and rescue to victims in an ever changing dynamic setting like big cities. resQ identifies victims and volunteers from social network data and quickly schedules rescue parties with an adaptive learning algorithm which does two key functions. Firstly, it identifies people that are trapped and in need of help as well as volunteers and rescue parties offering help. Secondly, it optimizes the volunteers' rescue strategy in a complex time sensitive environment. This reinforcement learning algorithm uses a heuristic function to speed up the training process by reducing the action-state space by prioritizing a set of actions over others. The paper sets up this Multi-Agent Reinforcement Learning as a standard seven tuple Markov Game $G =< N, S, A, P, R, \gamma >$. Here the Agents are the volunteers and rescue parties that total to N rescuers. State $st \epsilon S$ is the state si of agent i at time t. Action set is the set A and a discount factor γ is used to quantify the importance of immediate rewards versus future rewards, as seen in a standard Markov Decision Process. $P : S \times A \rightarrow [0, 1]$ is the transition function describing the probability of moving between states. R is a reward function $Ri \in R = S \times A \rightarrow (-\infty, \infty)$ which is a feedback from the environment when a volunteer takes action at a state. resQ uses a model free Q-learning algorithm along with a heuristic function that reduces the state space which in turn reduces the exploration steps required by the algorithm, hence the name resQ. [18] Extracts data from Twitter using Machine Learning during a disaster. The Tweets generated during the time of a natural disaster is analysed using data analytics and natural language analytics methodologies to gather useful information. Burst Detection is employed to monitor Twitter feeds worldwide to detect an increased activity and it raises an alarm for immediate action when an anomaly is detected like a surge in Tweets or hashtags. Tweet Filtering deals with filtering out the large quantity of irrelevant Tweets from the relevant ones. Support Vector Machines and Naïve Bayes classifiers were used to classify the Tweets as a part of Tweet Filtering. The features used were word bigrams and word frequencies, length of the words, user mentions, hashtag frequency, Tweet reply frequency by other users and whether a tweet is retweeted. Online Clustering was employed to group similar Tweets into what was assumed to be event specific topics. This paved way for Tweets extraction during the time of a crisis. [19] Shows how Social Media information was exploited during the Haiti Earthquake. It discusses the pros and cons of analytics using croudsourced information at the time of a crisis. It elaborates on the limitations and dangers that must be considered to make this an effective tool to mobilize response accurately and securely. It suggests that geotags extracted from Tweets need not be accurate and the data from Tweets can be widely exag- gerated that affects the priority level of that Tweet. Moreover, fraud reports with malicious intent could also be a challenge for the rescue teams. It also discusses the repercussions of making the data public - both positive and negative. Finally, it discusses the scalability and safety of such crowdsourced applications along with privacy.

[2] undertook a project as part of CrisisNLP to annotate Tweets that came from multiple disaster events and compiled a dataset upon which the proposed models in this paper are built. In addition to data collection and annotation, word vector representations [14] were learned using fifty two million crisis tweets. Support Vector Classifiers, Na¨ıve Bayes and Random Forest Classifiers are applied on the annotated Tweets as baselines for other works on the dataset. [20] Applied Convolutional Neural Networks for classifying disaster tweets collected in [2] by the same authors. The nine labels annotated previously was merged into two for a binary classification setting and it was observed that Convolutional Neural Networks per- formed better in the classification task. Google embeddings and CrisisNLP embeddings were used along with both RNN and CNN based models. The usage of pretrained embeddings resulted in an increased model accuracy compared to the models that learned the embeddings from scratch as the training progressed using the randomly initialized, trainable embeddings layer. [21] Classified Tweets using Bidirectional LSTM and CNN based models with pretrained Word Embeddings from CrisisNLP [2]. The paper evaluates four tweet classification models on CrisisNLP dataset and obtained similar results which shows that domain independent word vector embeddings like GloVe may be used in place of domain-specific word vector embeddings specifically with Bidirectional LSTM where the results showed the highest accuracy of 62.04%. It was also noted that this substitution of general-purpose word embeddings only tends to improve the model in the case of Long Short Term Memory based recurrent neural networks as it's accuracy was not impressive when compared to the model accuracy seen in convolutional neural networks that employed domain specific crisis embeddings. **CNN for Text Classification:** As opposed to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

recurrent neural networks being a standard for processing text, very deep convolutional neural networks were first applied in text classification in [22]. This works in the character level and uses small convolution and pooling operations. The accuracy of this model improved with depth, which ended up with 29 convolutional layers. It was seen that it outperformed state of the art convNets.
For all this to work, there is a need to automatically detect disaster related Tweets first. [23] deals with the Habagat flooding of Metro Manila in 2012. This was a binary classification which classified the Tweets as informative or uninformative. Naïve Bayes and Support Vector Machine Classifiers were used for this task and the SVM classifier performed significantly better than the Naïve Bayes classifier. These methods were initially used for event detection and for separating related Tweets from unrelated Tweets that circulate social media. Unsupervised methods were also used for topic modelling as seen in [24] which uses clustering methods to automatically detect and cluster different topics in Twitter. It was called ClusTop as it clusters topics. Clustop was able to automatically determine the number of topics and did not require a large number of hyperparameters. It works by constructing word network graphs by using Bigrams, Trigrams etc.

3. PROPOSED DEEP LEARNING MODEL

3.1 System Architecture: Our overall goal is to provide a tweet classification system which could be deployed in the backend of a web application that could deliver relevant tweets to different disaster response teams dealing with different types of aid. A tweet like "I have a hundred food packets for those in need in the downtown area" will not be relevant for the army which is carrying out the rescue for stranded individuals. However, the very same tweet will be welcome news for the people who are at the temporary lodgings set up by the response force. Hence, such a tweet should go to the category related to donations, which will in turn be used by the volunteers in charge of donations.

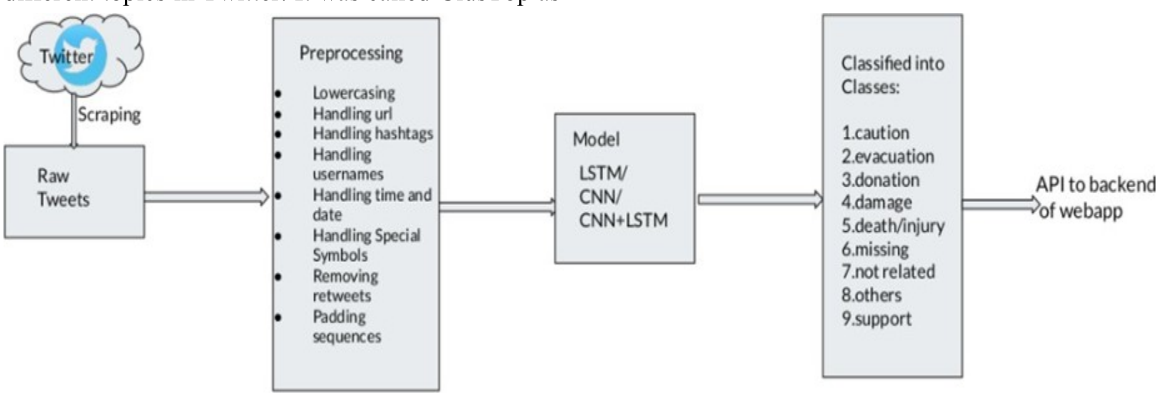

Figure 3.1: System Architecture

The proposed system's architecture is illustrated in Figure 3.1. The tweet processing pipeline is explained in the following steps.

3.1.1 Tweet Preprocessing:

The general steps involved in preprocessing text for modeling are not sufficient in preprocessing tweets. The major reason is that Tweets, or microtexts in general are smaller in size. Additionally, a Tweet could contain a large amount of 'anomalies' which usually do not occur in formal text which are seen in articles, essays and novels. These 'anomalies' that are common in microtexts are entities like URLs, hashtags, usernames that do not contain any meanings, smileys, RTs (retweets), timestamps etc.

3.1.1 Label Encoding

The class labels given as a text string for each of the Tweets had to be converted into a one hot encoded scheme with a field for each of the nine classes. The LabelEncoder found in sklearn.preprocessing was used to convert the string class name into an integer number ranging from 0 to 8. The output of the encoding scheme was as follows:
Printing the encoding scheme
0: warnings and advisories
1: evacuation and Rehabilitation
2: volunteering and donations
3: damage to infrastructures and utility
4: loss of life or injury

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5: person missing/found or trapped
6: irrelevant information
7: other useful information
8: sympathy and emotional support

The output of the encoding is shown below.

```
[[0. 0. 0. ... 0. 0. 1.]
 [0. 0. 0. ... 0. 1. 0.]
 [0. 0. 1. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 1.]
 [0. 0. 0. ... 1. 0. 0.]
 [0. 0. 0. ... 0. 1. 0.]]
```

3.1.1 Label Encoding

The class labels given as a text string for each of the Tweets had to be converted into a one hot encoded scheme with a field for each of the nine classes.
The LabelEncoder found in sklearn.preprocessing was used to convert the string class name into an integer number ranging from 0 to 8. The output of the encoding scheme was as follows:

Printing the encoding scheme

0: warnings and advisories
1: evacuation and Rehabilitation
2: volunteering and donations
3: damage to infrastructures and utility
4: loss of life or injury
5: person missing/found or trapped
6: irrelevant information
7: other useful information
8: sympathy and emotional support

4.2.1 Datasets
Imran et.al in [2] compiled a total of twelve datasets from twelve disasters. Three of them were related to respiratory diseases as oppossed to natural disasters. Due to a difference in domain as well as labelling scheme which was completely different from their natural disas- ter counterparts, the datasets with Tweets related to respiratory diseases were discarded. The list of datasets used are:

1) Pakistan Earthquake 2013
2) California Earthquake 2013
3) Chile Earthquake 2013
4) Hurricane Odile Mexico 2014
5) India Floods 2014
6) Pakistan Floods 2014
7) Phillipines Typhoon 2014
8) Cyclone Pam 2015
9) Nepal Earthquake 2015

Furthermore, paid workers labelled the data into one of 9 different categories which are:

I. Injured or dead people: Reports of casualties and/or injured people due to the crisis
II. Missing, trapped, or found people: Reports and/or questions about missing or found people
III. Displaced people and evacuations: People who have relocated due to the crisis, even for a short time (includes evacuations).
IV. Infrastructure and utilities damage: Reports of damaged buildings, roads, bridges, or utilities/services interrupted or restored.
V. 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.
VI. Caution and advice: Reports of warnings issued or lifted, guidance and tips.
VII. Sympathy and emotional support: Prayers, thoughts, and emotional support.
VIII. Other useful information: Other useful information that helps understand the situation.
IX. Not related or irrelevant: Unrelated to the situation or irrelevant

For this research work, all the data concerning each of the disasters were clubbed together so that the model will be independent of the type of disaster. The number of classes was kept as nine itself and hence the problem became a multi class classification problem with nine classes. The preprocessing steps were performed, and duplicates were dropped. The ninth class - ‘Not related or irrelevant' was also included as the model is also required to filter out unwanted tweets.

3.3 Models

A total of six neural network based models were built - with two variants each for an LSTM based model, a CNN based model and a combination of LSTM and CNN based model. Each of the models, as mentioned, comes in two variants. One which uses a pretrained embeddings layer for word embeddings from CrisisNLP [2] and the other which makes use of an embedding layer whose weights are trained during the course of model training. The pretrained weights of the embeddings layer is frozen while the other is learned from scratch, which is initialized at random.

3.3.1 LSTM

The model architecture for the LSTM based Recurrent Neural Network with learned embeddings is illustrated in Figure 4.2. In the learned embeddings variant, the embedding dimension is set as 128. In the case of the pretrained embeddings, the CrisisNLP pre- trained embeddings of 300 dimensions per token is used as seen in Figure 3.2. A total of 64 LSTM cells are present in the first LSTM layer which is having the hyperbolic tangent as activation and sigmoid as recurrent activation. This is then masked by a dropout of 0.3 for regularization. This layer undergoes Batch Normalization and is masked additionally by a dropout of 0.3. After this single layer of one directional LSTM, 64 dense neurons, activated by relu and initialized by glorot uniform is used. The bias is initialized with zeros. This layer also undergoes Batch Normalization and dropout to check covariance shift and prevent overfitting. The final layer is an output layer of 9 neurons - one for each class, wrapped by softmax activation.

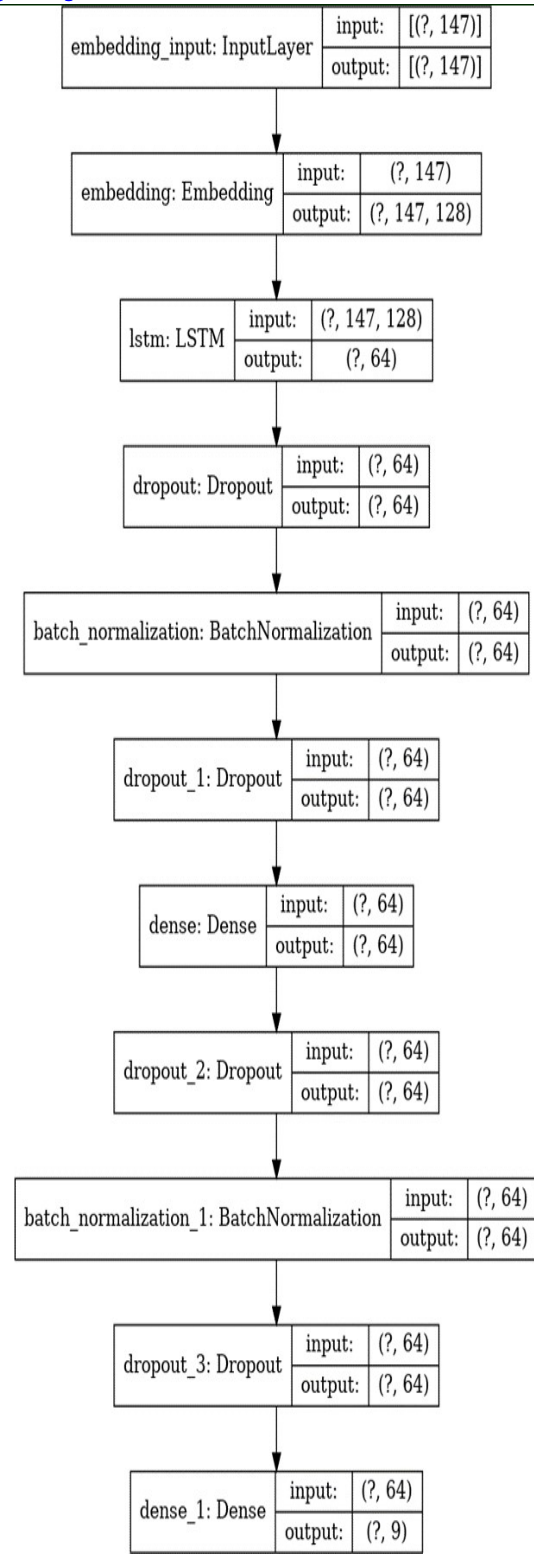

Figure 3.2: LSTM Based Recurrent Neural Network With Embeddings Learned From Scratch.

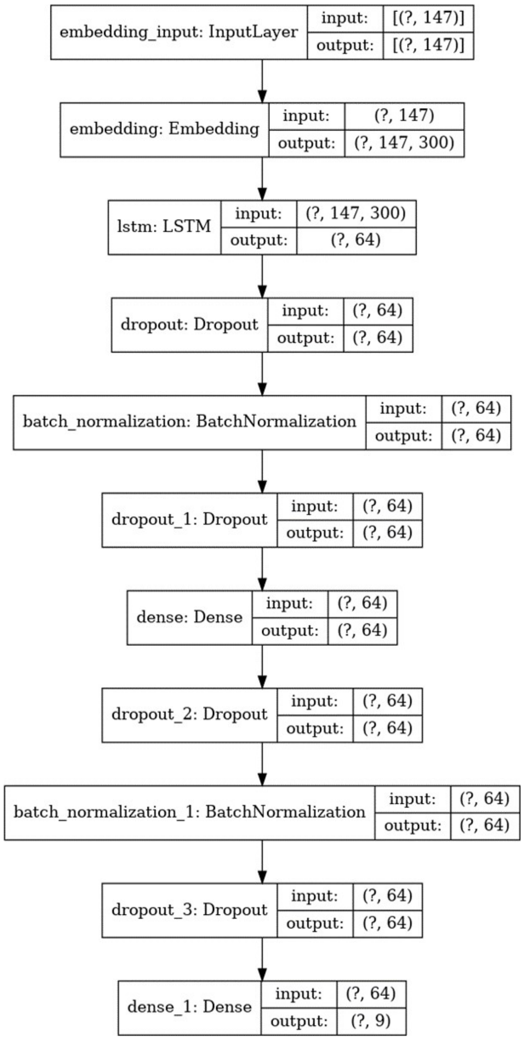

Figure 3.3: LSTM Based Recurrent Neural Network With Pretrained Crisisnlp Word2vec Embeddings.

3.4 CNN

The Convolutional Neural Network based model with pretrained embeddings is illustrated in Figure 4.5. In the illustrated pretrained embeddings variant, the word embedding di- mension is kept as 300. If the model is of the other variant where the embeddings are learned from scratch, the embedding dimension is kept at a low 100 dimensions per token as seen in Figure 3.4. A 1-D Convolution layer follows the embeddings layer. It has got 96 neurons. The kernel size is kept as low as 3 since the tweets being dealt with are short in length. The filter takes a stride of 1 token every time. The weights are initialized with glorot uniform initialization and the activation function used in this layer is relu. The output of the convolutional layer is then subjected to a global max pooling across 1 dimension. Two dropout masks of drop probability of 0.2 is used along with Batch Normalization. This output is then passed on to a dense layer of 96 neurons which is initialized with glorot uniform and with the same activation function of ReLu. Further layers of dropout masks of 0.2 are used for generalization. Finally, a dense layer of 9 neurons wrapped with a SoftMax layer is used for the output.

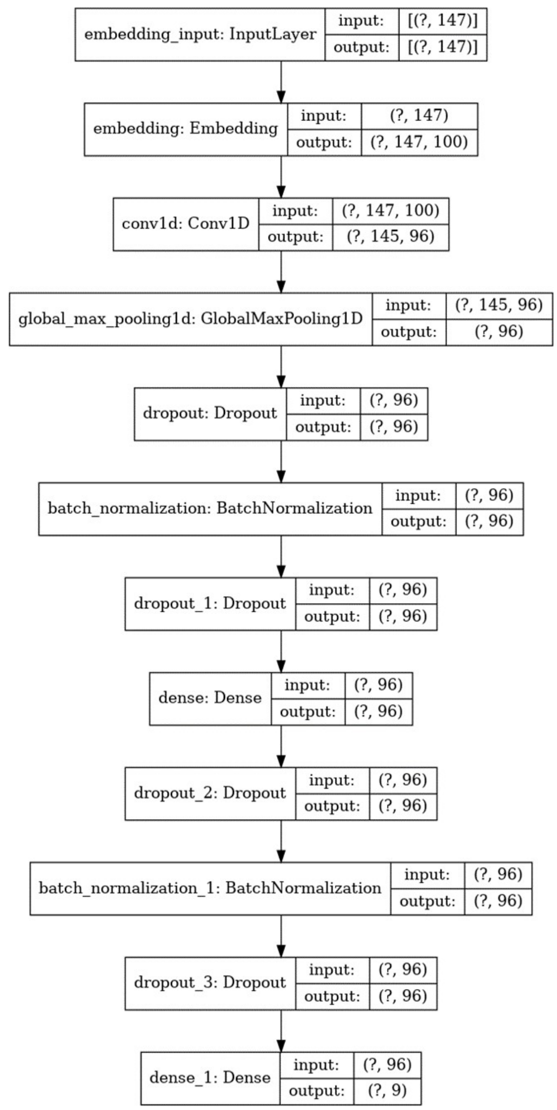

Figure 4.4: Convolutional Neural Network With Embeddings Learned From Scratch.

3.4 CNN-LSTM

Both the variants of this model will feed the output of a convolution layer to an LSTM layer. The pretrained variant of this model is illustrated in Figure 4.7. If the variant learns the word embeddings from scratch, the embedding dimension is set to 96 as seen in Figure 4.6. The pretrained embeddings come in 300 dimensions per token. A 1 dimensional convolution of 32 filters and a kernel size of 5 is used here with a stride length of 1. The activation is kept as relu. A max pooling layer of pool size 3 is used following the convolution. The output of this is **then passed on** to an LSTM layer of 32 recurrent neurons initialized as glorot uniform. The bias is initialized to zero. Dropout of 0.5 and Batch Normalization is used in this layer. This is then connected to a dense layer of 32 neurons which is initialized as glorot uniform. Here also the activation is kept as ReLu. Finally, for drawing the outputs, 9 neurons are wrapped in SoftMax activation.

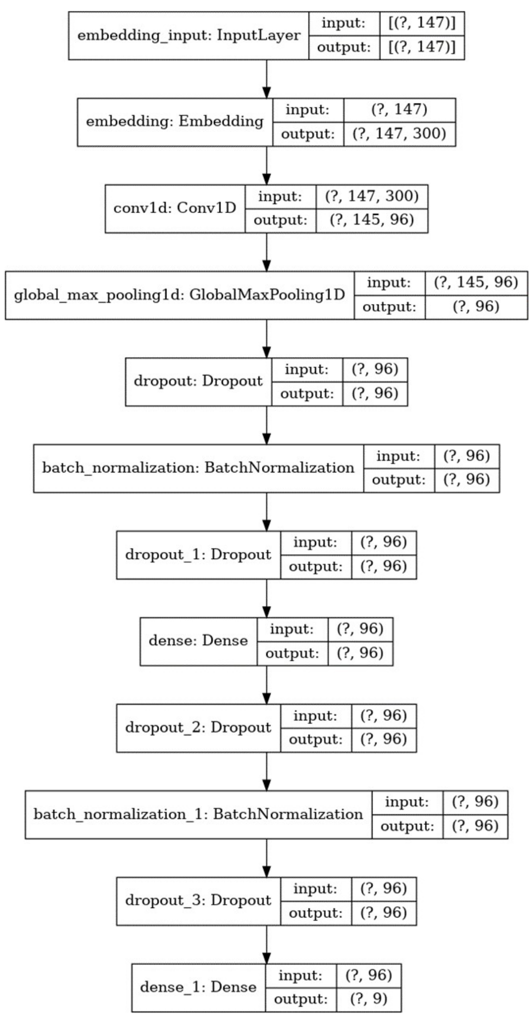

Figure 3.5: Convolutional Neural Network With Pretrained Crisisnlp Word2vec Embeddings.

Note that in all the six figures of the models, the question symbol on the first element of the tuple that defines the shape of the layers denote the input size which is the batch size you feed into the network, and is not a part of the network.

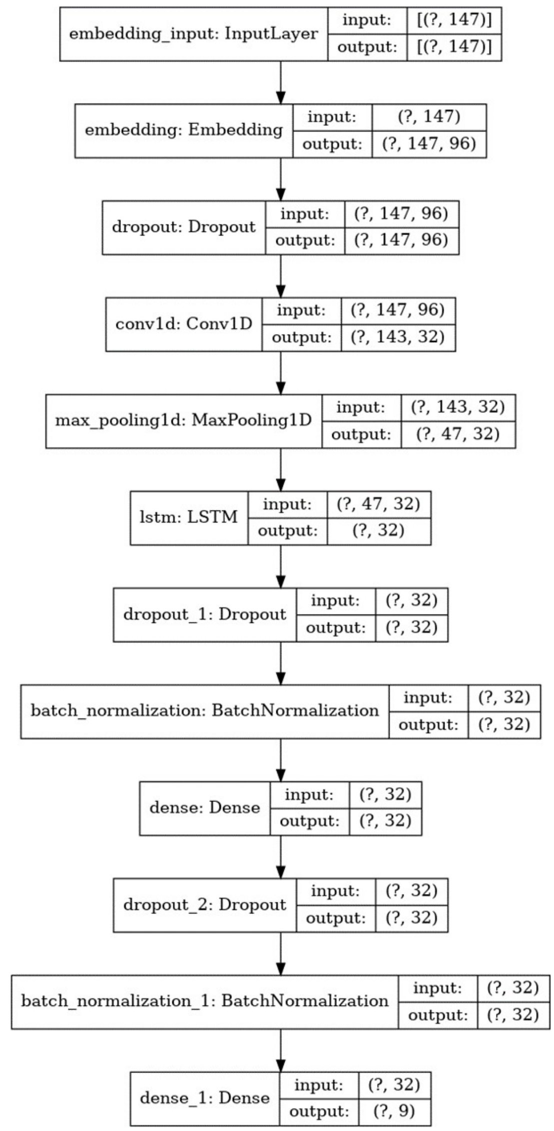

Figure 3.6: CNN-LSTM Based Neural Network With Embeddings Learned From Scratch.

3.5 Model Training

3.5.1 Optimizer, Loss and Metrics.

All the models were trained with Adam[9] as optimizer with the default parameters for β1, β2 and ϵ. Since this is a classification problem, the loss

Journal of Theoretical and Applied Information Technology
15th December 2023. Vol.101. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

function used here is categorical cross entropy from the output layer which is wrapped by SoftMax. For evaluation, the metrics chosen were accuracy, recall, precision, and f1-score. To train the models, the pre- processed data, in batches were loaded into the models with padding for different sequence lengths of the individual tweets. In all the models, the batch size was kept as a constant 128 tweets per batch and the number of epochs varied from model to model. The validation split was done in the ratio of 3:1.

4.5.2 Training History

The summary of the training process for all the six models is illustrated in Figure 3.8. The number of epochs trained is varied automatically based on monitoring the model error and stopping early as the error spikes.

5. **Results and Discussions**

In this section, we present the results obtained by all the six models which were built and trained.

5.1 Comparison of the Models

The cross-validated accuracy, precision, recall and f1-scores given by all the six models are summarized below:

Table 1 Comparison Of Results

Sl.N	Model	Accuracy	Precision	Recall	f1
1	LSTM	64.74	67.01	63.03	64.95
2	CNN	68.33	68.79	67.17	67.97
3	CNN-LSTM	67.35	74.21	60.94	66.94
4	LSTM + Embeddings	70.79	79.78	59.80	68.35
5	CNN + Embeddings	72.32	75.11	70.21	72.57
6	CNN-LSTM + Embeddings	70.82	76.93	64.68	70.27
7	CNN + Embeddings [21]	-	-	-	61.38
8	Bi-LSTM + GloVe [21]	-	-	-	62.04

It may be noted from Table 1 that the convolutional neural network based model with pretrained domain specific CrisisNLP word2vec embeddings are showing the best model performance in terms of accuracy and f1 score. It shows a better f1-score compared to the existing work. [21] which produced an f1 score of 61.38 for their CNN model using Crisis embedding and an f1-score of 62.04 for their Bidirectional LSTM model using generic GloVe embeddings.

5.3 Discussion

We can see that during the event of a crisis, data which may or may not be related to the crisis will be flowing in bulk through microtext channels like Twitter. Leveraging this could mean a better outcome for a lot of different people affected by the disaster. However, this is all under the assumption that the communication channels - especially the internet is minimally affected and will be accessible to the public during such an event. In such a case, the usage of language classification tasks to prioritize disaster management will be a faster way to provide relief than manually sorting the tweets into different levels of urgency.

6. **CONCLUSIONS**

The proposed model's architecture gives a method to simplify the process of prioritizing the delivery of disaster relief. Furthermore, it eliminates the need to manually sort the data and categorize them. This means that disaster relief, when automated in part, can be much faster and hence wider reaching than fully manual methods. Out of the six models which were built, the Convolutional Neural Network based model which uses domain specific Crisis Embeddings was the best with a much better f1 and accuracy scores. Moreover, since the domain specific unsupervised information was already effectively captured by the word embeddings, which was trained on a much larger, unsupervised twitter corpus, the supervised training did not require a large number of epochs to start converging. The CNN model which was built proved to be a stable model. It performs better than the existing work [21] done in 2019 using Bidirectional LSTMs. The use of neural models has an added advantage of automatic feature extraction when compared to other techniques like Support Vectors and Logistic Regression. Finally, the model was trained on a comparatively smaller dataset. For better results, there is a need for more domain specific supervised data - data which, unfortunately can only be obtained in the case of a natural disaster. Let us hope that mankind is more prepared in the times of crisis and are able to efficiently leverage technology like Deep Learning to save lives. Stay safe.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Future Scope: The proposed model could be enhanced by using the latest State of the art language mod- els like BERT (Google.Inc), ERNIE(Baidu/paddlepaddle) etc., and can be explored if the addition of a SOTA language model increases the model accuracy and if it does so, at what cost. Also, even though the models which were built were wrapped with nine SoftMax activated neurons for multiclass classification, it sometimes makes sense to just detect the most urgent class like "Caution or advice" for releasing urgent news to the public. This problem could be converted to a binary classification for this purpose and a broadcasting application could be developed that extracts relevant news and/or warnings and circulate it to the masses through either a standalone application or a bot that posts them on social media. This model could be enhanced as an online model which could learn as it works post deployment. Moreover, techniques like Reinforcement Learning could be employed as shown in [17] to automatically schedule volunteers to victims during a disaster. For this, we could convert the problem into a Multi-Agent Reinforcement Learning standard as a Markov Game.

REFERENCES:

[1] I. Shklovski, M. Burke, S. Kiesler, and R. Kraut, "Technology adoption and use in the aftermath of hurricane katrina in new orleans," American Behavioral Scientist, vol. 53, no. 8, pp. 1228–1246, 2010.

[2] M. Imran, P. Mitra, and C. Castillo, "Twitter as a lifeline: Human-annotated twitter corpora for nlp of crisis-related messages," 2016.

[3] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[4] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," vol. 15, 01 2010.

[5] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann ma- chines vinod nair," vol. 27, 06 2010, pp. 807–814.

[6] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," Journal of Machine Learning Research - Proceedings Track, vol. 9, pp. 249–256, 01 2010.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human- level performance on imagenet classification," IEEE International Conference on Com- puter Vision (ICCV 2015), vol. 1502, 02 2015.

[8] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, pp. 1929–1958, 2014.

[9] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," CoRR, vol. abs/1412.6980, 2015.

[10] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in Shape, contour and grouping in computer vision. Springer, 1999, pp. 319–345.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recur- rent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555, 2014.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word repre- sentations in vector space," 2013.

[15] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed represen- tations of words and phrases and their compositionality," in Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, ser. NIPS'13. Red Hook, NY, USA: Curran Associates Inc., 2013, p. 3111–3119.

[16] Z. Yang, L. H. Nguyen, J. Stuve, G. Cao, and F. Jin, "Harvey flooding rescue in social media," 2017 IEEE International Conference on Big Data (Big Data), pp. 2177–2185, 2017.

[17] L. Nguyen, Z. Yang, J. Zhu, J. Li, and F. Jin, "Coordinating disaster emergency response with heuristic reinforcement learning," 2018.

[18] J. Yin, S. Karimi, A. Lampert, M. Cameron, B. Robinson, and R. Power, "Using social media to enhance emergency situation awareness," in Twenty-fourth international joint conference on artificial intelligence, 2015.

[19] H. Gao, G. Barbier, and R. Goolsby, "Harnessing the crowdsourcing power of social

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

media for disaster relief," IEEE Intelligent Systems, vol. 26, no. 3, pp. 10–14, 2011.

[20] D. Nguyen, K. A. A. Mannai, S. Joty, H. Sajjad, M. Imran, and P. Mitra, "Robust classification of crisis-related data on social networks using convolutional neural networks," 2017. [Online]. Available: https://aaai.org/ocs/index.php/ICWSM/ICWSM17/paper/view/15655

[21] R. ALRashdi and S. O'Keefe, "Deep learning and word embeddings for tweet classifi- cation for crisis response," 2019.

[22] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional net- works for text classification," 2016.

[23] B. Parilla-Ferrer, P. Fernandez, and J. IV, "Automatic classification of disaster-related tweets," 12 2014.

[24] K. H. Lim, S. Karunasekera, and A. Harwood, "Clustop: A clustering-based topic modelling algorithm for twitter using word networks," 12 2017.

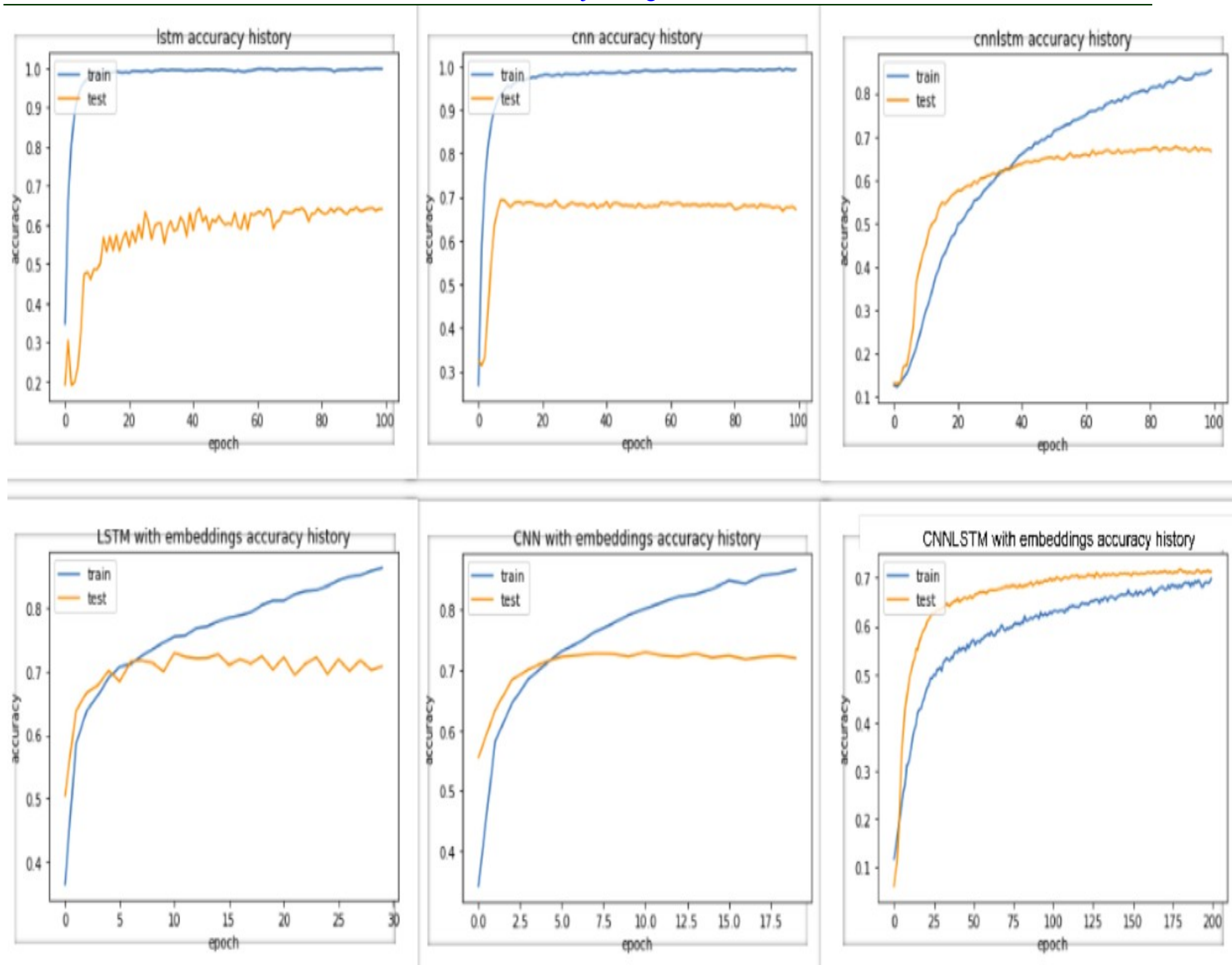

Figure 3.8: Training Summary Of All Six Models