Utkal University Journal of Computing & Communications,
Vol.1, Issue. 2(2023) *e-ISSN:2584-203X*

Enhancing Sentiment Analysis with Explainable AI: A Comparative Study of LSTM, GNN, and Capsule Networks

Indradev Sawa[1], Soumya Sahoo[2], Mamatarani Das[3], Prachi Priyanka[4], Arpit Anand[5], Shashwati Jha[6]
Department Of CSE, C.V. Raman Global University
mamataparida2005@gmail.com

Abstract: Sentiment analysis is vital for applications like social media monitoring and customer feedback analysis. While deep learning models such as Long Short-Term Memory Networks (LSTMs) and Graph Neural Networks (GNNs) achieve high accuracy on large datasets, their lack of interpretability remains a challenge. This paper proposes a hybrid approach combining advanced deep learning models with Explainable AI (XAI) techniques to enhance transparency without sacrificing performance. An LSTM model integrated with SHAP (SHapley Additive exPlanations) achieved 76% accuracy on a dataset of 7,613 tweets, providing insights into word- level contributions to sentiment predictions. Future work includes extending this framework with GNNs and Capsule Networks using XAI tools like GNNExplainer and LIME to capture complex relationships and hierarchical structures, ensuring both accuracy and interpretability for real-world applications.

Keywords: LSTM, GNN, Capsule Networks

1. Introduction

Sentiment analysis is an important step in many applications like social media monitoring, customer feedback analysis, disaster response. In recent years, sentiment analysis for short sequence of text has gained a lot of importance in both the business and public organizations, especially with the explosion of unstructured textual data on platforms such as Twitter. Traditional approaches to sentiment analysis statistically classify sentiment through machine learning-based techniques and have shown success in classifying sentiment correctly, but they do not provide interpretability and transparency in their predictions.

Noteworthy breakthroughs in deep learning modeling, especially LSTM, GNN, and CapsNet, have successfully dealt with complex data architectures including sequential text, graph, and hierarchical structures in textual elements. While they achieve great results on tasks such as sentiment classification, they can muddy the water — due to their black box nature, it often becomes hard to explain why a certain prediction has been made. The inability to interpret and explain the decision-making of these models in specific contexts can pose challenges in terms of trust and accountability, especially in high-stakes applications, such as disaster reponse or policy-making, that require insight into the model's reasoning process.

In this work, we respond to such challenges by proposing a hybridization approach with necessary innovative and advanced deep learning models along with Explainable AI (XAI) techniques in our proposed research study to increase the accuracy and accuracy of the sentiment analysis systems. This work specifically delves into three

✉ mamataparida2005@gmail.com

unique deep learning models: LSTM, GNN, and CapsNet, with each model uniquely designed to extract sentiment information from text data in varying facets. To address the trade-off between model performance and interpretability, we combine these models with state-of-the-art explainable artificial intelligence (XAI) methods including SHAP, LIME, and GNNExplainer, to obtain interpretable insights as to which underlying features drive these predictions.

This work aims to advance so called explainable sentiment analysis systems, namely in comparing with the literature, systems that not only achieve good performance in the classification of sentiment, but also provide explanations for their predictions that are meaningful. We seek to encourage transparency, ethical use of AI, and achieve higher levels of user trust in sentiment analysis applications by doing this. By offering a comparative analysis of these models on a disaster tweet classification dataset, this work shows that hybrid approaches (wherein deep learning is coupled with XAI) can provide high performance while retaining interpretability, and thus help making them suited for real-world applications.

2. Methodology

Sentiment analysis and topic modeling are two of the most important techniques that come under the umbrella of natural language processing (NLP) which has undergone a sea-change over time. On the other hand, topic modeling aims to extract the potential topics from a corpus, while sentiment analysis intends to determine the emotional content of the text. This shift from traditional rule-based systems to advanced machine learning and deep learning algorithms was fueled by the explosive growth of large corpus of text data and the performance boost brought by modern computational power in both domains.

Early sentiment analysis systems were largely rule-based and relied on a hand-collected lexicon of sentiment words to classify text as positive, negative, or neutral. Although these systems were simple to implement, they could not fully capture the complexity of language. For example, they could not handle sarcasm, irony, and ambiguity, and could misclassify the mood of a sentence based on the surface meaning of individual words. An example of such a limitation is the sentence “Oh, great job hacking into the system! The word “excellent” could be misinterpreted as positive, even though it is essentially a negative emotion [1]. In addition, early systems did not account for differences in emotion intensity, and words such as “good” and “excellent” were perceived as the same thing, although they had different emotional weights. This shortcoming emphasized the need for more sophisticated and context-aware methods.

Sentiment analysis underwent a major change with the introduction of statistical machine learning techniques. And so instead of handcrafted rules, performance driven data learnings [2] which methods like Naïve Bayes, Support Vector Machines (SVM) and Logistic Regression. These models represented text by using features like word frequencies, n-grams, or term-document matrices, and used pre-labeled datasets to learn patterns. For instance, a Naïve Bayes classifier can evaluate how likely a sentence is to be positive or negative depending on the count of specific words in the sentence. Although they were able to generalize and thus performed better than rule-based systems, these statistical models fell short on more complicated linguistic phenomena, e.g., polysemy (i.e. one word having multiple meanings) and pragmatic subtleties, e.g. negations. The meaning of the phrase “not bad,” which has a positive connotation, was also frequently misclassified due to its syntactic complexity.

The advent of deep learning, and subsequently recurrent neural network architectures (RNN), and even more specifically Long Short Term Memory (LSTM) networks brought a whole new paradigm to sentiment analysis, successfully addressing many of the challenges posed by previous systems [3]. As LSTMs are proved to be capable for processing sequential data and modeled on the complex long- term dependencies, it was definitely an appropriate approach for analyzing the text where the context and order of word are essential to provide accurate meaning of the sentence [4]. For instance, in a sentence like "The film was a bit slow but had a fantastic ending," LSTMs could successfully identify the difference between the negative sentiment expressed by "slow" and the positive sentiment expressed by "fantastic", and would classify the sentiment as positive. LSTMs also addressed the issue of vanishing gradients, which plagued traditional RNNs by using gating mechanisms to retain important information over long sequences.

The transformer-based models, such as Bidirectional Encoder Representations from Transformers (BERT), marked a breakthrough in sentiment analysis by leveraging self- attention mechanisms allowing the model to take the right and the left context into account for a target word. Even though LSTMs read a sequence at a time, transformers read the entire sentence at once, looking for relations between all words in the context. This approach to comprehending language in a bidirectional context enables BERT and similar models to address ambiguous or nuanced text more effectively, resulting in more accurate sentiment detection. For instance, in the phrase "The product was surprisingly good for its price," BERT might find the positive sentiment in the adverb "surprisingly," perhaps without a keyword search. With the rise of transformer models, sentiment analysis has made significant strides by improving the classification accuracy across multiple contexts, including customer reviews and social media [5].

Along with the development on sentiment analysis, Topic modelling emerged as one of the most popular modelling technique used to find hidden factors withing the large text corpus. LDA (Latent Dirichlet Allocation), one of the earliest and most commonly used techniques, is a probabilistic model of documents as mixtures of topics, where each topic is a distribution over words. LDA is particularly useful for finding hidden thematic structures in large datasets and does not require labelled data. For instance, if we were analysing a collection of news articles, we would typically see dominant themes around politics, economics, and sports which would be identified by LDA. But, LDA has its shortcomings, especially regarding complex relationships between words. It is also based on the assumption that (words within) a topic are independent given the distribution of words, which is rarely the case in actual text, and therefore often fails to capture complex semantic relationships [6].

In an effort to overcome the limitations mentioned, some researchers are proposing hybrid models that integrate the best of both the neural networks and the traditional topic modelling techniques. These hybrid approaches utilize word embeddings such as Word2Vec or GloVe to model the semantic similarity between words, improving the topic discovery process's quality. Those models are usually an integration of word embedding and LDA or other topic model [7]. Word embeddings have been integrated into LDA models, which tend to outperform LDA on finding topics in realistic datasets, such as studies tracking sentiment towards public policy issues during the COVID-19 pandemic and capturing themes such as healthcare, vaccine distribution, and economic policies.

In marketing, customer service, public health, policy analysis, and many more areas, sentiment analysis, and topic modelling have had far- reaching applications. Sentiment analysis is commonly used in social media to assess public opinion on election outcomes, so that the authorities can learn about the public's position on issues such as brand perception and crisis management. Analysis of social media posts during the COVID-19 pandemic is one such case which enabled to observe the public angle and the trend with the changes [8].

Instrumental in uncovering thematic trends in online discussions, such as identifying key concerns about healthcare resources or economic impacts during the pandemic. Sentiment Analysis and Topic Modeling: These techniques are applied to understand the sentiments and opinions expressed in vast amounts of text data, such as on Twitter, news articles, and blogs, providing valuable insights to empower decision- making by businesses, governments, and other stakeholders. However, despite superb and remarkable progress in sentiment analysis and topic modeling, there are still challenges. One of the issues is very much the quality of the input data — particularly around social media. The data produced by these platforms is often noisy, biased, and unstructured, which leads to incorrect or skewed outcomes. As an illustrative example, extreme views on controversial topics often dominate the online discussion, leading sentiment analysis (which uses opinion as examples for positive and negative) to be skewed positively or negatively, while more moderate views are ignored. Moreover, the use of slang, emojis, and abbreviations in social media language makes it challenging for conventional NLP models. While engineered for social media data, tools such as VADER (short for Valance Aware Dictionary and Sentiment Reasoner), which helps to parse informal text, work effectively with specific forms of communication, they still struggle to keep pace with the language emerging in social networks and alternate mediums of online communication.

A major hurdle is the computational burden of sophisticated models. Deep learning architectures such as LSTMs and transformers need significant computational resources and large labeled datasets to be trained, making them perhaps unreachable for small organizations or researchers with limited infrastructure. In addition to this, these models are prone to over fitting, especially when trained on small or noisy datasets. This necessitates the use of regularization practices and robust evaluation protocols to nip overfitting in the bud; however, it complicates the modeling process. Between this and the need for real-time analysis in fast-moving contexts (e.g., social sentiment can change quickly) creates even more obstacles. For example, in the case of large events or crises, insights need to be timely, while the latency of processing large volumes of text data may limit how well one can track emerging trends or respond quickly enough.

To summarise, sentiment analysis and topic modeling has come a long way, from simple rule-indicator systems to more complex systems of deep learning and hybrid methods. Text analysis has benefited from these techniques, leading to the development of applications across domains, including marketing, social media analysis, and policy-making. Yet issues of data quality, computational capabilities and real-time processing persist — signifying the necessity for ongoing innovation and development in these spheres. These techniques are being used extensively and can prove to be insightful and help with decision-making in several fields.

3. Objective

This research aims to improve sentiment analysis by combining advanced deep learning models with explainable AI (XAI) techniques. Long Short-Term Memory networks (LSTMs), Graph Neural Networks (GNNs) and Capsule networks are three of the major families of models that this study focuses on, as they each provide their own unique advantages in case of complex and hierarchical textual data. Dual-Stage Attention-Based Encoder-Decoder Network for Business Review Generation — LSTMs are used for modeling sequence of data. Some examples are GNNs for processing graph-structured input, say users and their preference on social media, and Capsule Networks that can better understand text where hierarchical relationships might be important, e.g. detecting sarcasm or negation. One of the main challenges associated with deep learning is interpretability, which can be tackled using XAI methods.

LIME (Local Interpretable Model-Agnostic Explanations): It explains individual predictions by approximating the local area around the example. This prevent research intends to develop sentiment analysis systems that provide high performance with high transparency through the incorporation of high performance deep learning models with the XAI. The far reaching goal is making sure that with this kind of models tools, systems, or applications where they could be useful like analyzing customer feedback, designing marketing strategies or decision-making processes or formulating policies, they are fairly interpretable and human understanding, thereby leading to trust in AI systems and keeping them within the drape of accountability.

4. Data Processing

This dataset includes various tweets regarding disasters, where the eaim is to identify whether an individual tweet is connected to a disaster (1) or not (0). This data preprocessing phase consists of a few essential steps that also aid in cleaning and converting the text into a structured format that can be trained by a machine learning model. The training data then undergoes text cleaning, where special characters, punctuation, numbers, and URLs which do not contribute to the task and can add noise to the model are removed. For example, symbols such as “@”, “#” and numbers are not considered as they do not add any value to the classification. Normalization In the text cleansing, normalization is the next step; that is, convert all the text into lower case so it will be standardized, and also avoid case sensitivity, for instance, when you have "Flood" and "flood," both the same but different in case.

After cleaning, the next step is tokenization and stemming. Tokenization involves splitting the tweet into individual words or tokens. For example, the tweet "The earthquake is devastating" would be tokenized into ["The", "earthquake", "is", "devastating"]. Stemming and lemmatization are applied to reduce words to their base forms. Stemming removes prefixes and suffixes to standardize words (e.g., "running" becomes "run"), while lemmatization converts words to their dictionary forms (e.g., "better" becomes "good"). These processes help in reducing the variability in word forms and allow the model to focus on the core semantics of the text.

With this stop-word removal is conducted. Stop-words like "the", "is", "and", etc., are common and do not provide useful information for classification. Excluding these makes the model pay attention to the relevant contents of the tweet and makes it memory efficient. The first step would be preprocessing the text data. The text is then processed using the TF-IDF (Term FrequencyInverse Document Frequency) method and transformed, arriving at a word vector. This method measures how important a word in a tweet belongs to the whole dataset. Tags or other references to locations that are common within a specific tweet but rare throughout the dataset are weighted heavily. This allows capturing unique terms which are likely to indicate if a tweet is related to a disaster or not. The output is the tweets in a vector space which can then be fed into the machine learning algorithms.

3. Model Architecture

The architecture of the model greatly affects the success of tweet classification, as the model needs to be able to capture the relationships and patterns within the tweet dataset. Therefore, three advanced architectures were used in the tweet data, since it includes specific characteristics such as sequential dependencies, spatial hierarchies, and complex relationships between words. The architectures are Enhanced-Graph Neural Network (GNN), Bi-Directional Long Short-Term Memory (BiLSTM), and Capsule Network (CapsNet).

The Enhanced Graph Neural Network (GNN), which is used to model the complex interrelationship between words of a tweet In contrast to many traditional models, GNNs adopt a graph-based perspective: each word in a tweet is treated as a node, with the relationships between words encoded as edges. The further GNN architecture reflects these interdependencies through information passing along the graph structure, effectively modeling for example the influence of one word on another. The tweet text is vectorized using the TF-IDF method, and these feature vectors are passed in through a series of fully connected layers. Each fully connected layer is followed by a batch normalization layer to enhance the speed and stability of the training and to avoid vanishing or exploding activations. The last layer gives a binary classification output, predicting if the tweet is disaster related (1) and if it is not (0). To add non-linearity after each fully connected layer, Leaky ReLU activation is used, which allows the model to learn more complex patterns. Dropout is used both during training to prevent overfitting, with a rate of 0.3. This architecture is proficient in memorizing complex patterns and correlation, which exist in the tweet data.

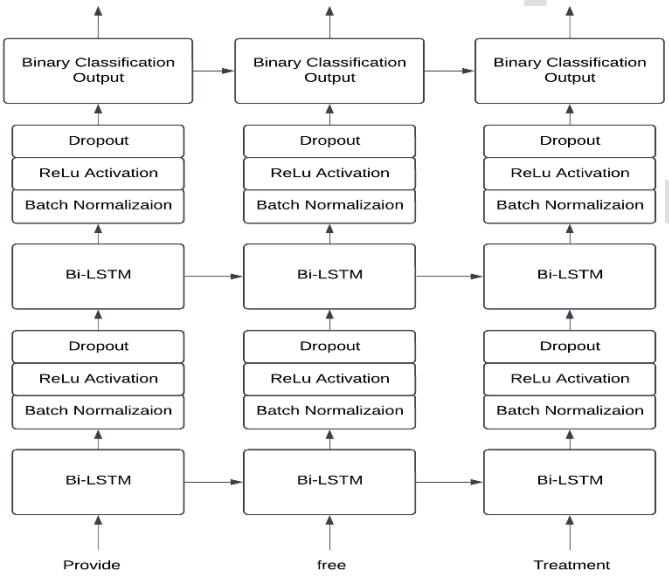

Fig. 1. Structure of the Bi-LSTM designed for sentiment clas sification.

Another robust architecture that can be employed for tweet classification is the Bidirectional Long Short-Term Memory (BiLSTM) network shown in Figure 1. Long short-term memory (LSTM) is a type of recurrent neural network (RNN) capable of learning long-term dependencies. What makes BiLSTM different is that it reads the tweet text forward and backward. Since a tweet is a sequence of words, the model takes into consideration the surrounding context from both past and future words which helps it understand the meaning of the tweet better. For instance, in the tweet “The earthquake is devastating,” BiLSTM can use the context of both the word "earthquake" and the word "devastating" to comprehend the entire context. This BiLSTM output layer directly turns into a fully connected layer which finally yields the classification. The LSTM layer is then dropped out with a fraction of 0.5 to avoid overfitting. To protect against exploding gradients, a common problem for RNNs, MLP, and denser models, we also apply gradient clipping. Disaster tweet classification shines with BiLSTM approach, being able to regain correlation between prior and subsequent words due to the complexity of natural language.

A more recent innovation is the Capsule Network (or CapsNet) architecture, which tries to preserve the spatial hierarchies of data is shown in Figure 2. This is especially critical for tweet-classifying tasks where the position and

order of words in the tweet itself is vital to understanding its meaning. The first part of the algorithm is in the construction of capsules. In CapsNet, there is not single output neuron but grouped by capsules representing different characteristic features of the tweets. The convolutional layer that first receives input from the data is composed of features, such as individual words or phrases. The intermediate features are then fed to the primary capsule layer, where they are grouped into capsules. Each capsule refers to separate property of a tweet, such as sentiment or disaster relevant words like, “flood,” or, “earthquake.” The next step is apply the dynamic routing algorithm to weight the importance of each feature and then only the most relevant features are added to the final classification. Dynamic routing is beneficial as it allows the model to consider the most salient relationships between words, rather than giving equal consideration to all features. The output for each capsule is squashed with a nonlinear function that yields a probability-based representation for each tweet. Margin loss is utilized in place of traditional cross-entropy loss, supporting the model to make certain predictions. CapsNet architecture has proven to be efficient in learning the finer relationships between the words while retaining the spatial information of the tweet, which makes it an apt choice for disaster tweet classification.

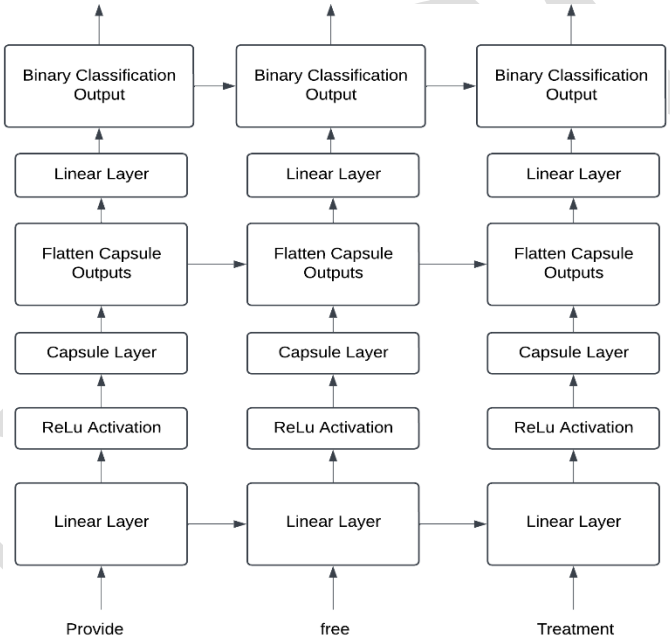

Fig. 2. Structure of the Capsule Network designed for sentiment classification.

Collectively these three models, Enhanced GNN, BiLSTM and CapsNet were used to address the various dimensions of the tweet data. The GNN models word mutual dependencies, while BiLSTM accounts for sequential context and CapsNet preserves the spatial hierarchy of words. The results show the ability of the system to classify disaster-related tweets efficiently due to the combined models, despite complex relationship and variations of structure. These models were further refined with LIME (Local Interpretable Model-Agnostic Explanations) for better interpretability of the given inputs prediction. LIME enables the models to describe how they reach their decisions by finding the words in the tweet that influence most the classification. This enhances the interpretability of the models and also offers an overview of the workings of the classification process, leading to easier adjustments and improvements of the model performance.

Table 1. Performance of Different Classifiers

Model	Result Matrix			
	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)
GNN	77.36	72.76	71.19	71.96
LSTM	77.54	77.56	66.56	71.64
CapsNet	76.30	71.43	73.96	72.67

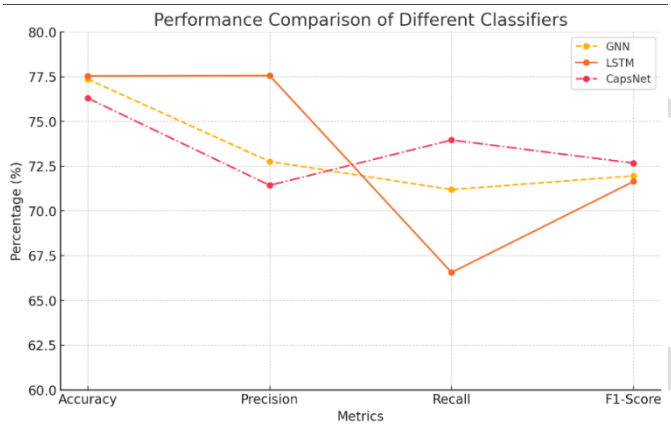

Fig. 3. Performance Comparison of Different Model.

4. Implementation And Results

The models were evaluated across important metrics such as accuracy, precision, recall, and F1-score including Graph Neural Network (GNN), Capsule Network (CapsNet), and Long Short- Term Memory (LSTM) mentioned in Table 1. Finally, the LSTM model showed the best performance at an accuracy of 77.54%, compared to GNN with 76.36% and CapsNet with 76.30%. This indicates that temporal dependency captured by LSTM enables it to perform better than CNN in the correct classification of disaster-related tweets. For precision, the LSTM again outperformed with a value of 77.56%, indicating its capability in accurately classifying disaster-related tweets while reducing false positives shown in Figure 3. In contrast, GNN and CapsNet tended to over-predict disaster-related tweets with lower positive predictive values of 72.76% and 71.43%, respectively.

In terms of recall, the CapsNet model achieved the highest at 73.96%. This means that it was the best in terms of capturing all tweets relevant to disasters, but it sacrificed precision, as it is higher in false positives. Also, recall values for the GNN and LSTM models, 71.19% and 66.56% respectively indicate that these models missed some disaster-related tweets. This indicates a trade-off between recall and precision, where CapsNet favors recall but loses precision compared to LSTM, which has high precision but misses some tweets related to disaster. With respect to F1-score, CapsNet took the lead with a value of 72.67%, which reveals its good balance between precision and recall. Despite a lower recall, LSTM has an F1-score of 71.64%, while GNN had an F1-score of 71.96%, slightly higher than LSTM. These results emphasize the high performance differences for these models in various aspects of disaster tweet classification.

5. Discussion

The results reveals the differentiated strengths and weaknesses for each model's performance. The GNN demonstrates a balanced performance overall but struggled with recall, as it failed to identify a significant number of disaster-related tweets, leading to higher false negatives. The moderate precision in the model indicates that some susceptibility to false positives is present within the model's system, suggesting that while the GNN was able to capture contextual relationships in the data, it wasn't as effective in identifying all disaster-related content. The capsnet model achieved better recall, suggesting that it can effectively capture disaster-related tweets, while it has a trade-off of lower precision. It seems that CapsNet is hard to classify disaster-related content and over-predict unlike most models. Although this property may be desirable in contexts such as those where being able to label as many relevant tweets (in terms of disasters) as is humanly possible is importantly relevant like news aggregators, it could also foretell a large number of falsely marked tweets. The LSTM model, in comparison, performed exceptionally in terms of accuracy and precision, neither misclassifying non-disaster tweets as disaster-related nor over-predicting disaster-related tweets. However, given its lower recall score, the LSTM misclassified some disaster-related tweets (missed them), which means it does not perform well for used-cases, where capturing every possible disaster-related tweet is critical, even at the cost of having some false positives. These findings indicate that GNN is effective and well-balanced across classes, Capenet is sensitive to disaster-related content, and the LSTM model shows high precision at a lower recall. Thus each model can be optimized well for a particular use case or strengths can be combined from the different models for achieving a better performance in disaster tweet classification tas.

6. Conclusıon

In conclusion, this study is that, we compared three advance deep learning models in detail against each other i.e Graph Neural Network (GNN), Capsule Network (CapsNet) and Long Short-Term Memory (LSTM) to classify disaster related tweets. The analysis showed that every model performed based on the task in a different way. Highest accuracy and precision rate was obtained by LSTM model, which was able to classify disaster-related tweets without overpredictions. The ability of this model to capture temporal dependencies which are crucial for understanding the context of tweets significantly contributed to its superior performance. The ability of this model to capture the temporal dependencies important to the context in the understanding of tweets contributed greatly to its performance. On the other hand, CapsNet, which can capture the similarity and sequential relationship between words in a text, had the highest recall out of all models, meaning it was the most successful at finding disaster-related tweets. But this increased recall was accompanied by a decrease in precision, as the model was also prone to over-predicting disaster-content, resulting in more false positives. Though the overall performance of the GNN model was more balanced, it had a much lower recall, failing to correctly classify a large number of disaster-related tweets, leading to a much larger amount of false negatives. Despite this, the GNN exhibited moderate precision, suggesting that it was relatively effective at minimizing false positives.

References

1. H. Jelodar, Y. Wang, R. Orji, and S. Huang, "Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach," IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 10, pp. 2733–2742, Oct. 2020.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
2. P. Durga and D. Godavarthi, "Deep-Sentiment: An Effective Deep Sentiment Analysis Using a Decision-Based Recurrent Neural Network (D-RNN)," IEEE Access, vol. 11, pp.
3. Aygün, B. Kaya, and M. Kaya, "Aspect Based Twitter Sentiment Analysis on Vaccination and Vaccine Types in COVID-19 Pandemic With Deep Learning," IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 5, pp.
4. X. Yu, M. D. Ferreira, and F. V. Paulovich, "Senti-COVID19: An Interactive Visual Analytics System for Detecting Public Sentiment and Insights Regarding COVID-19 From Social Media," IEEE Access, vol. 9, pp.
5. P. Gupta, S. Kumar, R. R. Suman, and V. Kumar, "Sentiment Analysis of Lockdown in India During COVID-19: A Case Study on Twitter," IEEE Transactions on Computational Social Systems, vol. 8, no. 4, pp. 939–949, Aug. 2021.
6. G. Yang, H. He, and Q. Chen, "Emotion-Semantic-Enhanced Neural Network," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 27, no. 3, pp. 531–540, Mar. 2019.
7. H. T. Phan, N. T. Nguyen, V. C. Tran, and D. Hwang, "Improving the Performance of Sentiment Analysis of Tweets Containing Fuzzy Sentiment Using the Feature Ensemble Model," IEEE Access, vol. 8, pp.
8. T. Wang, K. Lu, K. Chow, and Q. Zhu, "COVID-19 Sensing: Negative Sentiment Analysis on Social Media in China via BERT Model," IEEE Access, vol. 8, pp.