

---

# Emergency Tweet Categorization and Prioritization

---

Tanishk Raj, Vaibhav Rana, Adam,  
Anubhava Srivastava\* and Himanshu Sharma

*School of Engineering and Technology, Department of Computer Science and  
Engineering, Sharda University, Greater Noida, Uttar Pradesh, India  
E-mail: anubhavacse@gmail.com*

*\*Corresponding Author*

Received 08 January 2025; Accepted 01 May 2025

## **Abstract**

The “Emergency Tweet Categorization and Prioritization System” is an all-encompassing solution aimed at reducing the workload of emergency service departments by classifying and prioritizing tweets. The system utilizes Google’s BERT model to increase the effectiveness of the rescue process. In natural language processing and text classification, numerous projects have attempted tweet categorization; however, our project specifically addresses prioritizing tweets for emergency departments. The BERT model is optimized to distinguish the authenticity of the tweets and label the tweets which are related to the fire department, police, and medicals. The train dataset has been balanced through the process of oversampling and undersampling to be able to run smoothly and highly accurately. The model has been able to post an 87% accuracy ratio in predicting authenticity. Some of the future directions include time- and distance-based prioritization of tweets, expanding the system to emergency calls, and designing real-time monitoring and alerting mechanisms.

**Keywords:** Natural language processing, undersampling, oversampling, real-time monitoring, BERT.

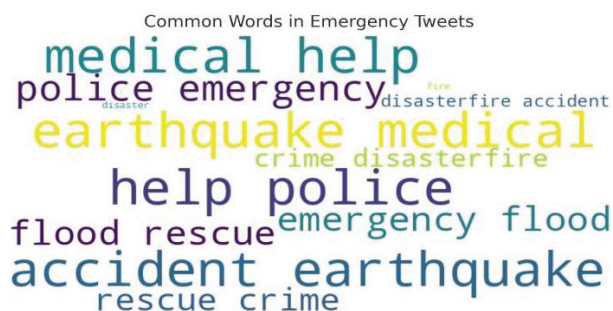
*Journal of Mobile Multimedia, Vol. 21\_3&4, 455–474.*

doi: 10.13052/jmm1550-4646.21347

© 2025 River Publishers

## 1 Introduction

Good communication is important in crisis situations to facilitate prompt response and assistance. In times of crisis, the dynamics of communication become more complex with interactions involving real people as well as attackers with ill motives [1, 2]. Against this, technology such as BERT provides promising solutions. Designed by Google, BERT takes advantage of pre-training on massive corpora, e.g., for sentence-level and token-level processing tasks [2]. Text classification is a critical function in natural language processing (NLP), which allows the categorization and organization of text data, an imperative for information retrieval and comprehension. In emergency communication, new technologies have been highly impactful. For instance, an IVR solution, ODIN IVR, which is an interactive one, was developed and deployed, significantly helping rescuers and illustrating the importance of advanced communication systems in emergency times [3]. Supervised machine learning techniques that had previously been utilized in labelling the tweets as either newsy or not recognized much dependably on the reliability of data from social network sites, like Twitter [4]. Conventionally based NLP solutions have been in deep investigation for feature space enhancement, deep techniques like DBN, RNN, and CNN [5]. Transfer learning methods by fine-tuned pre-trained word vectors in tandem with a CNN showed astonishing accuracy on sentence classification, with only minor consideration tuning required [6]. Model-specific fine-tuned vectors expedite this process. Even though the mainstream of neural networks has made massive advances, it is slowly being overtaken by transformers. In 2018, Google introduced BERT, an encoder-decoder architecture-based algorithm that revolutionized text classification with pre-training on large corpora of language.



**Figure 1** Word map for types of emergencies.

The first and foremost purpose behind this idea is to increase the classification accuracy without increasing the preprocessing effort. Contextual understanding of the word, rather than just word by word understanding, has defined a new benchmark for NAT procedures. The factors that fuel the shift from neural networks to BERT-based models in text classification are a variety. First, BERT has large-scale pre-trained language representation, contributing to better generalizability and performance on downstream tasks. Second, BERT-like transformers wield greater capability to process contextual information, making them better suited for complex language tasks. Finally, BERT-based models also tend to obviate the necessity of a heavy preprocessing effort, an aspect that decelerates the development cycle and brings up the faster delivery of NLP applications. We found that the system exploits the ameliorated ability of BERT augmenting the text classification, particularly in the field of emergency communication. With BERT, the system also aims to enhance accuracy and processing speed of vast quantities of text data for efficient information retrieval and decision-making in crisis situations. This transition from the conventional neural networks to transformer models is a breakthrough in NLP, with BERT being at the forefront of establishing new standards for text classification and other language processing tasks.

## **2 Literature Survey**

In recent years, the proliferation of social media has provided new avenues for emergency response systems, particularly through the classification and prioritization of emergency-related tweets. Several studies have explored various machine learning and deep learning techniques for text classification, sentiment analysis, and emergency response optimization. Yoon Kim [6] investigated the application of Convolutional Neural Networks (CNNs) for sentence classification tasks. By leveraging pre-trained word vectors, the study demonstrated that a simple CNN architecture could achieve excellent results across multiple benchmarks with minimal hyperparameter tuning. This research highlights the potential of CNNs for text classification tasks, which is relevant for emergency tweet categorization. J. Laksana and A. Purwarianti [7] focused on classifying authoritative sources within Indonesian Twitter data for governmental purposes in Bandung. Their research aimed to identify credible information sources, providing valuable insights into social media analysis tailored for government applications. This work aligns with emergency tweet categorization by emphasizing the importance

of source credibility in emergency response. Y. Li et al. [8] addressed feature extraction from requirements documents using neural networks. Their study aimed to enhance accuracy and automation in the extraction process, contributing to advancements in software analysis and engineering. The techniques proposed in their research can be adapted to improve feature extraction in emergency-related tweets, reducing manual intervention and increasing efficiency. A. Alagarsamy and Ruba S. Kathavarayan [9] proposed a modified deep belief network for classifying large datasets. Their work emphasized improving classification performance and efficiency in handling extensive data, which is crucial for processing high volumes of emergency tweets during crises. S. Lai et al. [10] introduced recurrent convolutional neural networks (RCNNs) for text classification. Their approach combined recurrent and convolutional networks to capture both local and sequential information, improving text classification performance. This technique can be beneficial for categorizing emergency tweets that require both contextual understanding and sequential dependencies. P. Liu et al. [11] investigated recurrent neural networks (RNNs) in a multi-task learning framework for text classification. Their research demonstrated that RNNs could effectively handle multiple related tasks, improving overall efficiency. This framework can be adapted for multi-label classification of emergency tweets, enabling better prioritization. Ashish Vaswani et al. [12] introduced the Transformer architecture in their seminal paper, "Attention Is All You Need." The Transformer model, relying entirely on self-attention mechanisms, has become foundational in NLP by enabling parallelization and reducing training times. Its application in emergency tweet classification can significantly enhance accuracy and processing speed. S. Prabhu et al. [13] explored multi-class text classification using BERT-based active learning. Their study leveraged the power of BERT models combined with active learning strategies to improve classification accuracy, particularly in scenarios with limited labeled data [14, 15]. BERT-based approaches are highly relevant for prioritizing emergency tweets. A. Çelikten and H. Bulut [16] applied BERT to Turkish medical text classification, demonstrating the model's adaptability to different languages and specialized domains. This highlights BERT's potential [17, 18] for emergency tweet classification across different languages and contexts. A. Murarka et al. [19] utilized RoBERTa for classifying mental illnesses based on social media data. Their research showcased the potential of advanced language models in identifying mental health issues from online platforms. This technique can be leveraged to detect distress or urgent medical [20, 21] emergencies in tweets. D. Warner et al. [22] characterized call prioritization

times in a Police Priority Dispatch System, providing valuable insights into the efficiency of emergency response systems. Their findings emphasize the importance of timely call handling, which is crucial for designing tweet prioritization frameworks. K. Torlén Wennlund et al. [23, 24] investigated emergency medical dispatchers' experiences in managing emergency calls. Through qualitative interviews, they identified key challenges and decision-making processes, offering recommendations for improving emergency medical services. Understanding dispatcher experiences can inform the development of an emergency tweet prioritization system that aligns with real-world response protocols. The aforementioned studies provide a solid foundation for developing robust emergency tweet categorization and prioritization frameworks. Leveraging advanced NLP techniques such as CNNs, RNNs, Transformers, and active learning can enhance the efficiency and accuracy of emergency response systems, ensuring timely and effective intervention.

### **3 Methods**

#### **3.1 Approach Overview**

Training and fine-tuning are the first two processes in the BERT paradigm. The model is trained using unlabeled data from a variety of pre-training activities during the pre-training phase. Then comes the fine-tuning, where pre-trained parameters are initialized in the BERT model and each parameter is re-tuned using labeled data from downstream tasks that are essentially the same as those performed by any number of wide-learned, domain-agnostic neural nets. Besides being the first model to be truly sensitive to word order, BERTBASE (the base version of BERT) is also a remarkable architecture, containing 12 layers of stackable transformer components. Since transformer networks are attention-based, the BERT model can perform 12-way parallel attention, that is, each of 12 components can, in essence, be looking at different places in the input text. Moreover, BERT uses a bidirectional architecture that allows the model to make use of both left and right context when encoding words.

#### **3.2 Masked Language Model (Masked LM)**

A pretraining task used in the development of Google's BERT Model (Bidirectional Encoder Representations from Transformers). During training, the masked tokens are replaced with a special token denoted as "[MASK]." The

model has learned to understand the context surrounding the masked token and predict the most probable token that should replace it.

### **3.3 Next Sentence Prediction (NSP)**

Another pretraining task used by BERT is Next Sentence Prediction (NSP). This task, quite simply, involves training the model to predict whether two given sentences follow each other logically within a single document, or if they're just random pairs. The model

gets two sentences and learns to determine if the second one is the one that follows it in the document. (And if not, it must figure out which of its many possible next sentences is the right one.) In training, we categorize the sentences as either “IsNext” or “NotNext.”

### **3.4 Fine-tuning of BERT**

The pre-trained model is further trained on a task-specific, labeled dataset during the second phase of BERT training, known as fine-tuning. The procedure of text classification involves the addition of a task-specific classification layer to the pre-trained BERT model. The whole model is then fine-tuned on the labeled data, tuning the weights of the pre-trained BERT layers as well as the additional task-specific layer. The importance of fine-tuning in text classification is that it allows the model to learn to fit the unique features and demands of the classification task. By fine-tuning BERT over a labeled data set, the model learns task-specific patterns and adjusts its parameters to improve prediction accuracy. Contextual representations developed during pre-training serve as a strong foundation, and fine-tuning adjusts the model for individual classification tasks like sentiment analysis, topic classification, or spam classification.

## **4 High-Level Design**

### **4.1 Considerations**

In Figure 1, the overall constraints in developing the system are addressed, whereas the development processes, techniques, and methods used are explained. The flow of the work proposed includes some important steps. First, the Disaster Tweet Dataset is loaded, and data cleaning is performed, then the dataset is split into training and testing sets for appropriate evaluation. The next step is text data tokenization using the BERT tokenizer, with

tokens translated into the input formats BERT uses for compatibility. The pre-trained BERT model is loaded, and a classification layer added to the architecture to specialize it for the target classification task. The model is then fine-tuned through pre-trained initialization followed by training using the preprocessed data. To address the imbalanced dataset, methods like under-sampling or over-sampling can be used. The fine-tuned model is tested by predicting the testing set and is measured by different metrics, such as accuracy, precision, recall, and F1 score. New tweets are finally preprocessed following the same steps mentioned above and are fed into the fine-tuned BERT model to get the predictions for their classification.

## **4.2 General Constraints**

Collecting and pre-processing the dataset is the first challenge. Training the model using a small dataset to reach accuracy is another major challenge. Handling imbalanced datasets, in which class distribution is uneven, is another key constraint. The use of proper techniques is required to handle this imbalance and prevent biased model performance. Proper care should be taken during training to avoid overfitting and underfitting.

## **4.3 Development Methods**

The methodology employed in system development enables independence between the four modules. The development order was not significant as each module could be developed separately. The process entailed dividing each module into user stories depending on the needs of individual team members. On completion of every module, a test sprint was conducted to verify its usability. To counteract the drawback of an imbalanced dataset, oversampling and undersampling methods were employed. These methods increase the proportion of the minority class (oversampling) or decrease the proportion of the majority class (undersampling) to balance the dataset. Through the application of these methods, the dataset is put into balance to reduce the class imbalance problem and enhance model performance.

## **4.4 Programming Language**

Python is the language of choice for development because it has a rich library and framework ecosystem, which allows for rapid development and integration. Furthermore, to train the model to make classifications, we are using BERT, a cutting-edge transformer-based model. Python's ability to

allow rapid development and prototyping is demonstrated by its readability and simplicity, which allow for efficient coding and maintenance. Python's massive support community and large documentation also simplify solving problems and debugging. As far as the use of BERT in training the model is concerned, it has been outstanding in multiple natural language processing applications, especially text classification. BERT's ability to identify contextual information and process intricate language patterns makes it a prime candidate for dataset classification. By using pre-trained weights of BERT and fine-tuning the model over the target data, we will enhance accuracy and generalization.

#### **4.5 Paradigm of User Interface**

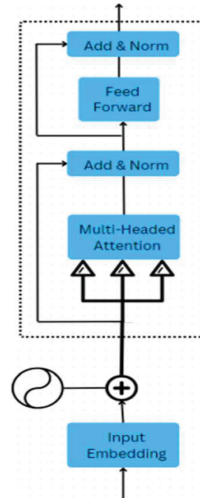
The interface model for labeling the dataset to classify it through BERT is employing Google Collab or Jupyter Notebooks. These systems provide an interactive coding platform integrating code cells along with helpful observation, allowing ease of code running and documentation. Users interact with the system in the notebook user interface, running code cells for organizing as well as training the dataset in BERT. The notebook interface supports an easy-to-use and interactive way of offering and organizing code, visualizations, and text descriptions, hence improving a fluid and effective workflow for dataset classification.

#### **4.6 Error Detection and Recovery**

The issue was encountered where the precision of the outcomes was not acceptable based on the unbalanced dataset. Certain techniques were hence applied in balancing the dataset to enhance the performance of the system. Techniques of under-sampling and over-sampling were applied to adjust class distribution to provide adequate representation for both minority and majority classes. By balancing the dataset, the ability of the system to detect and correct errors was improved, and more accurate and stable results were obtained.

#### **4.7 Data Storage Management**

The dataset exists in CSV form. Data storage management is managing this dataset in an optimized manner to process as much data with BERT as possible. The dataset is therefore being loaded from the CSV for this reason. By eliminating data that is irrelevant, the system focuses on relevant



**Figure 2** Transformer architecture used in natural language processing.

information, hence avoiding computational load and improving efficiency. This streamlined process guarantees that BERT runs on the dataset efficiently, optimizing its performance within the task of classification. Effective data storage management guarantees that the system utilizes the dataset optimally, resulting in effective and accurate outputs.

#### 4.8 System Architecture

The encoder of BERT model maps the input sequence to a continuous representation containing the information learned from the sequence. It includes a few layers, each of which has sub-modules that cooperate to process and transform the input data together. As shown in Figure 2, the sequence of input is passed through a word embedding layer, which gives each word within the sequence a vector representation, thus enabling learning of the model from numeric data. Secondly, positional encoding is added to the embeddings to provide information regarding the position of any given word in the sequence.

To do this, the input embeddings are converted into query, key, and value vectors. The dot product of queries and keys produces a score matrix, which establishes the relevance of every word paying attention to others. These are scaled and passed through a SoftMax function to derive attention weights. The weights are then utilized to calculate an output vector that aggregates information from all the relevant words in the sequence. Multiple attention



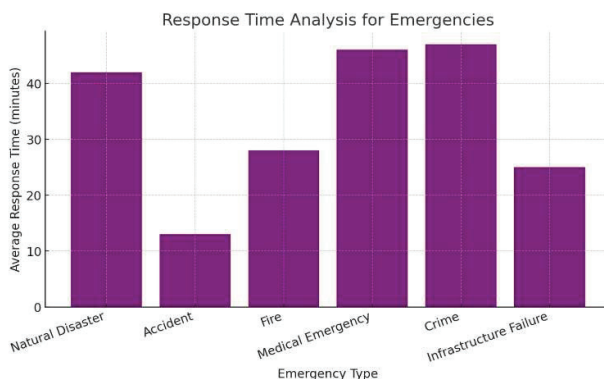
**Figure 3** BERT-based tweet classification system.

heads are employed to capture various relationships, providing a richer representation. Residual connections and layer normalization are applied to the output of the self-attention mechanism. Residual connections facilitate gradient flow during training, thereby improving optimization, while layer normalization stabilizes the network's behavior. The output is then fed into a point-wise feed-forward network, which comprises several fully connected layers with non-linear activation functions in between. This feed-forward network further processes the attention output, enhancing its representation. The BERT encoder stacks multiple layers of self-attention and feed-forward networks. The model can capture progressively complicated patterns and dependencies in the input sequence since each layer executes the same set of operations after receiving the output from the previous layer. The final output of the BERT encoder is a sequence of contextualized representations for each word in the input, encoding both the individual word's meaning and its relationship to other words in the sequence. This enables the model to capture rich semantic information and perform effectively on various downstream tasks.

## 5 Experimental Results

### 5.1 Evaluation Metrics

Evaluation metrics are fundamental for assessing various procedures and understanding algorithm behavior. Metrics enable the resolution of how algorithms perform. Some methods satisfy certain metrics. This system was evaluated based on how well the expectations were met in a given scenario. A classification model's performance is evaluated using a confusion matrix. It makes it easier to calculate different performance indicators by summarizing the expected and actual labels for a classification problem. True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) are the four main parts of the confusion matrix.



**Figure 4** Average response time (in minutes) for different types of emergencies.

## 5.2 Datasets

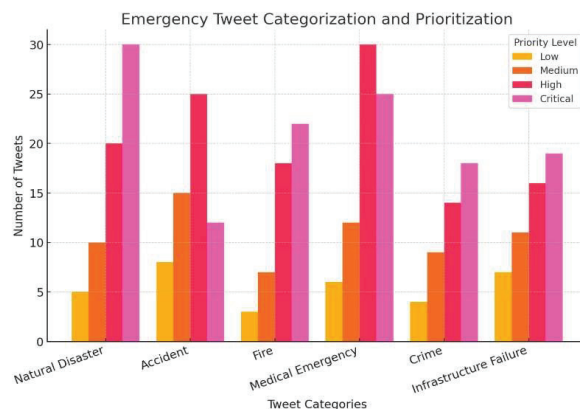
The ‘Disaster Tweets’ dataset, which has been verified by Kaggle, was employed to train the model. Location and keyword information, as well as over 11,000 tweets associated with disaster-related keywords such as “crash,” “quarantine,” and “bush fires,” are included in this file. The texts were painstakingly categorized to ascertain whether each tweet was a reference to a disaster event (e.g., a joke, a movie review, or something non-disastrous).

Additionally, the dataset was modified, and the tweets were manually categorized, as illustrated in Figure 3. Tweets relevant to police, fire, and medical emergencies were categorized accordingly. The number is a working example of this categorization. The data was divided into two subsets: training and testing. Seventy-three percent of the data was utilized for training, and twenty-seven percent was utilized for testing the model.

## 6 Evaluation & Results

### 6.1 Future Enhancements

**Time-and-Distance-Based Contextualized approach:** In order to prioritize tweets based on their vicinity to an incident scene, it will include a temporally and spatially based parameterization method. With real-time location tracking of the tweets, those closer to the emergency will receive higher priority to fast-track emergency response and resource allocation. **Integration of Emergency Calls:** Extending this system will include calls along with tweets. The development in this presence will involve the use of speech recognition



**Figure 5** Number of tweets related to different emergency categories.

software and natural language processing techniques, the result of which will classify emergency calls based on their urgency level. The surrounding emergency calls will make the emergency response more comprehensive and more inclusive. Real-Time Alert-and-Monitoring System: A real-time alert-and-monitoring module that constantly watches social media streams and other appropriate information streams for activity. This module should utilize sophisticated algorithms to identify and classify incipient emergencies in real-time. By promptly identifying and categorizing these emergencies, the system can deliver immediate alerts to relevant authorities, enabling them to respond swiftly and allocate resources effectively.

## 6.2 Evaluation Metrics

Evaluation metrics are principles that are used to test a variety of procedures and to comprehend the behavior of algorithms; these can be evaluated using metrics. Some strategies fulfill a portion of the metrics. This system was evaluated based on how many of the predictions worked perfectly in each situation. The efficiency of a classification model is assessed using a confusion matrix. It enables the computation of a variety of performance metrics by providing a summary of the predicted and actual labels for a classification problem.

The confusion matrix has four important components: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). From the confusion matrix (Figure 6), several performance metrics can be derived, including recall, precision, and F1 score.

id	keyword	location	text	target	Medical	Fire	Police	
0	0	ablaze	NaN	Communal violence in Bhainsa, Telangana. 'Ston...	1	0.0	1.0	1.0
1	1	ablaze	NaN	Telangana: Section 144 has been imposed in Bha...	1	0.0	0.0	1.0
2	2	ablaze	New York City	Arsonist sets cars ablaze at dealership https:...	1	0.0	1.0	0.0
3	3	ablaze	Morgantown, WV	Arsonist sets cars ablaze at dealership https:...	1	0.0	1.0	0.0
4	4	ablaze	NaN	"Lord Jesus, your love brings freedom and pard...	0	0.0	0.0	0.0
5	5	ablaze	OC	If this child was Chinese, this tweet would ha...	0	0.0	0.0	0.0
6	6	ablaze	London, England	Several houses have been set ablaze in Ngembib...	1	0.0	1.0	0.0
7	7	ablaze	Bharat	Asansol: A BJP office in Salanpur village was ...	1	0.0	1.0	1.0
8	8	ablaze	Accra, Ghana	National Security Minister, Kan Dapaah's side ...	0	0.0	0.0	0.0
9	9	ablaze	Searching	This creature who's soul is no longer clarent ...	0	0.0	0.0	0.0

Figure 6 Dataset used to train the model.

	precision	recall	f1-score	support
0	0.93	0.79	0.86	168
1	0.82	0.94	0.88	168
accuracy			0.87	336
macro avg	0.87	0.87	0.87	336
weighted avg	0.87	0.87	0.87	336

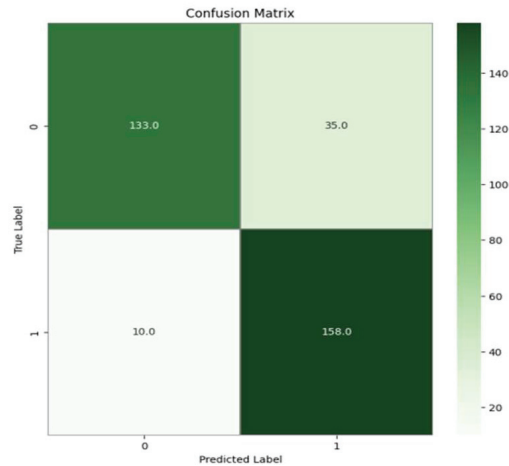


Figure 7 Confusion matrix.

### 6.3 Performance Analysis

This unit clarifies the experimental consequences of the scheme. The trained system successfully was tested against the random tweets present in the training dataset. The accuracy of the system was measured, and the data is presented in Figure 10.

```

Under_Sampling
Evaluation metric

11/11 [=====] - 15s 1s/step
      precision  recall  f1-score  support
0         0.93     0.79     0.86     168
1         0.82     0.94     0.88     168

accuracy      0.87
macro avg     0.87     0.87     0.87     336
weighted avg  0.87     0.87     0.87     336

Performance Analysis.
Out[10]: target
1      2071
0      2071
Name: count, dtype: int64
    
```

**Figure 8** Evaluation metrics achieved through undersampling.

```

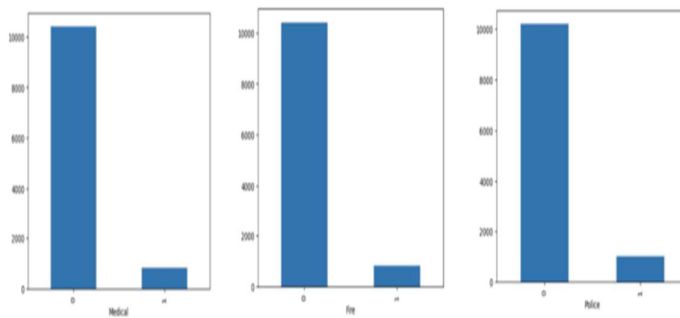
Over_Sampling
Evaluation metric

29/29 [=====] - 28s 772ms/step
      precision  recall  f1-score  support
0         0.96     0.91     0.94     739
1         0.68     0.85     0.76     168

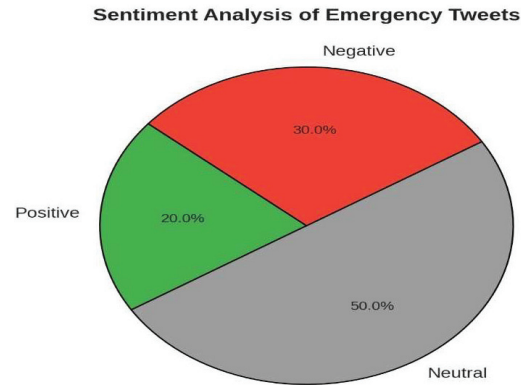
accuracy      0.90
macro avg     0.82     0.88     0.85     907
weighted avg  0.91     0.90     0.90     907

Performance Analysis
Random over-sampling:
target
0      9154
1      9154
Name: count, dtype: int64
    
```

**Figure 9** Evaluation metrics is achieved through oversampling.



**Figure 10** Imbalanced dataset for categorized tweets.



**Figure 11** Sentiments of tweets.

## 7 Conclusion & Discussion

The effective application of emergency tweet categorization and prioritization with a deployed modified Kaggle disaster tweet dataset proves the system's efficacy in correctly classifying tweets and prioritizing them according to their importance relative to various emergency categories. The process of natural language processing and machine learning methods uses the tweets' content to extract relevant keywords and patterns to classify the tweets. Moreover, the capability of the system to determine actual or non-actual emergencies is important in weeding out spurious or unnecessary information. This ensures that resources are properly directed, and emergency response efforts are targeted towards actual situations. In all, the successful deployment of this system has the potential to significantly improve emergency response efficiency. Of particular interest is that our model was able to achieve an accuracy of 87%.

## References

- [1] L. Hong et al., "Towards understanding communication behavior changes during floods using cell phone Data" in *Lect. Notes Comput. Sci. Soc. Info* 2018, S. Staab, O. Koltsova and D. Ignatov, Eds. *Social Informatics*, 2018(), vol. 11186. Springer, Cham
- [2] S. Skogevall et al., "Telephone nurses' perceived stress, self-efficacy and empathy in their work with frequent callers," *Nurs. Open*, vol. 9, no. 2, pp. 1394–1401, 2022 Mar. doi: 10.1002/nop2.889. PMID:34528768; PMCID:PMC8859069.

- [3] Lai, S. Xu et al., “Recurrent convolutional neural networks for text classification,” *Proc. AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.
- [4] B. Mocanu et al. “ODIN IVR-Interactive Solution for Emergency Calls Handling” *Appl. Sci.*, vol. 12, no. 21, p. 10844, 2022. doi: 10.3390/ap122110844.
- [5] C. Castillo et al., “Information credibility on twitter” in *Proc. 20th International Conference on World Wide Web (WWW ’11)*. New York, NY, USA: Association for Computing Machinery, 2011, pp. 675–684. doi: 10.1145/1963405.1963500.
- [6] H. Liang et al, “Text feature extraction based on deep learning: A review,” *EURASIP J. Wirel. Commun. Netw.*, vol. 2017, no. 1, p. 211, 2017. doi: 10.1186/s13638-017-0993-1.
- [7] Y. Kim, “Convolutional neural networks for sentence classification” in *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, 2014, 1746–1751. doi: 10.3115/v1/D14-1181.
- [8] J. Laksana and A. Purwarianti, *Indonesian Twitter Text Authority Classification for Government in Bandung*, 2014, pp. 129–134. doi: 10.1109/ICAICTA.2014.7005928.
- [9] Y. Li et al., “Extracting features from requirements: Achieving accuracy and automation with neural networks,” *IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, Campobasso, Italy, 2018, 2018, pp. 477–481. doi: 10.1109/SANER.2018.8330243.
- [10] A. Srivastava, S. Umrao, S. Biswas, R. dubey, and M. I. Zafar, “FCCC: Forest Cover Change Calculator User Interface for Identifying Fire Incidents in Forest Region using Satellite Data,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 7, pp. 948–959, 2023, doi: 10.14569/IJACSA.2023.01407103.
- [11] S. Lai et al., “Recurrent convolutional neural networks for text classification,” *AAAI*, vol. 29, no. 1, 2015. doi: 10.1609/aaai.v29i1.9513.
- [12] Srivastava, A. 2024. Temporal analysis of multi-spectral instrument level and surface reflectance data sets for seasonal variation in land cover dynamics by using Google Earth Engine. *Geodesy and Cartography*. 50, 4 (Dec. 2024), 162–178. doi: <https://doi.org/10.3846/gac.2024.20106>.
- [13] A. Vaswani et al., “Attention is all you need,” *Arxiv. /abs/1706.03762*, 2017.

- [14] A. Çelikten and H. Bulut, “Turkish medical text classification using BERT,” 29th Signal Processing and Communications Applications Conference (SIU), Istanbul, Turkey, vol. 2021, 2021, pp. 1–4, doi: 10.1109/SIU53274.2021.9477847.
- [15] A. Murarka et al., “Classification of mental illnesses on social media using Roberta” in Proc. 12th International Workshop on Health Text Mining and Information Analysis, 2021, pp. 59–68.
- [16] S. Prabhu et al., “Multi-class text classification using BERT-based active learning,” Arxiv./abs/2104.14289, 2021.
- [17] A. Çelikten and H. Bulut, “Turkish medical text classification using BERT,” 29th Signal Processing and Communications Applications Conference (SIU), Istanbul, Turkey, vol. 2021, 2021, pp. 1–4, doi: 10.1109/SIU53274.2021.9477847.
- [18] A. Srivastava and S. Biswas, “Analyzing Land Cover Changes over Landsat-7 Data using Google Earth Engine,” Proc. 3rd Int. Conf. Artif. Intell. Smart Energy, ICAIS 2023, pp. 1228–1233, 2023, doi: 10.1109/ICAIS56108.2023.10073795.
- [19] D. Warner et al., “Characterization of call prioritisation time in a Police Priority Dispatch System,” *Annals Emerg. Dispatch Resp.*, vol. 2, no. 2, pp. 16–22, 2014.
- [20] Sharma, Himanshu, Prabhat Kumar, and Kavita Sharma. “Recurrent Neural Network based Incremental model for Intrusion Detection System in IoT.” *Scalable Computing: Practice and Experience* 25.5 (2024): 3778–3795.
- [21] A. Srivastava, R. Dubey, and S. Biswas, “Comparison of Sentinel and Landsat Data Sets over Lucknow Region Using Gradient Tree Boost Supervised Classifier,” *Lect. Notes Networks Syst.*, vol. 730 LNNS, pp. 221–232, 2023, doi: 10.1007/978-981-99-3963-3\_18.
- [22] Sharma, H., Kumar, P., and Sharma, K. (2025). *Advanced Security for IoT and Smart Devices: Addressing Modern Threats and Solutions. Emerging Threats and Countermeasures in Cybersecurity*, 191–216.
- [23] H.J. Kim, *Feature Extraction of Text for Deep Learning Algorithms: Application on Fake News Dectection*, 2020.
- [24] S. Skogevall et al., “A survey of telephone nurses’ experiences in their encounters with frequent callers,” *J. Adv. Nurs.*, vol. 76, no. 4, pp. 1019–1026, 2020 Apr. doi: 10.1111/jan.14308:1111/jan.14308. PMID:31997365.

## Biographies



**Tanishk Raj** is pursuing the bachelor's degree in computer science & engineering from Sharda University 2021–2025. His research areas include artificial intelligence security, deep learning, professional writing. He has been serving as a writer for many projects. He has a keen interest in project management & content management.



**Vaibhav Rana** is a Bachelor of Technology (B. tech) student specializing in Computer Science and Engineering (CSE) at Sharda University (2021–2025). His research focuses on cybersecurity and its related fields, including cloud security, network protection, and ethical hacking. Passionate about safeguarding digital systems, Vaibhav explores innovative solutions in threat detection, data privacy, and secure software development. Through his research, he aims to contribute to advancements in cybersecurity and its real-world applications.



**Adam** is a Bachelor of Technology (B. Tech) student who specializes in Informatics and Engineering Science (CSE). His research focuses on natural language processing (NLP) and its respective fields, including machine learning, deep learning and calculation gardens. The AI-operated language model detects innovative solutions in emotional, adam-text therapy, emotional analysis and condensed AI. Through his research, their goal is to contribute to progress in NLP and its real applications.



**Anubhava Srivastava** is an Assistant Professor in the Computer Science Department at Sharda University, where he focuses on real time images, machine learning, and remote sensing. He earned his Ph.D. in AI/ML-based land use classification from RGIPT and brings a wealth of research experience in applying AI to geospatial analysis and environmental monitoring. His expertise covers a range of areas, including database management systems, artificial intelligence, remote sensing, and operating systems. Dr. Srivastava has a strong publication record in SCI/SCOPUS-indexed journals and conferences, with key studies on topics like forest transformation, vegetation indices, GIS-based road traffic noise mapping, and AI-enhanced environmental monitoring.



**Himanshu Sharma** is assistant at the Department of Informatics and Engineering Science at Sharda University, where he specializes in IoT security. He served his M.Tech. India, a degree from GBPUAT in India, and a concrete research foundation in cyber security, has received special attention to secure IoT ecosystems and analyze images in different applications and create innovative processing techniques. Currently, his research focuses on improving the safety of IoT devices and networks, while being in advanced imaging methods for real-time use. In addition to its research work, Himanshu is actively involved in technical workshops, education conferences and knowledge sharing forums, and ensures that it remains on top of the latest trends in cyber security and data view. Their final goal is to develop sophisticated solutions that help ensure our rapidly developed digital world.