Evaluating Bert Variants For Disaster-Related Information Classification: Identifying Situational Vs Non-Situational Data

Dattatray Sahebrao Shingate[1]*, Dr. Shyamrao Vasantrao Gumaste[2]
[1]Research Scholar, Department of Computer Engineering, Mumbai Education Trust's Institute of Engineering, Nashik, India.
[2]Professor, Department of Artificial Intelligence & Data Science, Mumbai Education Trust's Institute of Engineering, Nashik, India.
*E-mail: dattatrays_ioe@bkc.met.edu

Article Received: 10 May 2025, Revised: 17 June 2025, Accepted: 26 June 2025

Abstract

Natural language processing (NLP) plays a critical role in disaster management, particularly in distinguishing between situational and non-situational information shared during emergencies. This paper presents an evaluation of various BERT (Bidirectional Encoder Representations from Transformers) variants for classifying disaster-related information. The study focuses on identifying the most effective approach for accurately categorizing data into situational (relevant to disaster response) and non-situational (irrelevant or secondary) information. Several BERT models, including BERT-base, RoBERTa, and DistilBERT, are fine-tuned on a dataset containing real-world disaster communications such as social media posts, news articles, and emergency updates. We compare their performance based on metrics such as accuracy and processing time. Our findings highlight the strengths and limitations of each variant, with a particular focus on how each model handles the nuances of situational information during crises. The results demonstrate that RoBERTa outperforms other models in accuracy, while DistilBERT offers a faster alternative with minimal trade-offs in performance. This comparative analysis provides insights into selecting the optimal BERT model for real-time disaster response systems, ultimately contributing to faster and more accurate decision-making during emergencies.

Keywords: Disasters, BERT, RoBERTa model, DistilBERT, Lemmatization, POS tag, NLP, situational information, Event detection.

1. INTRODUCTION

In recent years, the increased frequency and intensity of natural disasters have emphasized the need for efficient and timely disaster response mechanisms. During such events, vast amounts of information are shared across multiple platforms, particularly social media, news outlets, and emergency communication systems. Extracting actionable, situational information from these sources can be a challenging yet crucial task for ensuring an effective disaster response. The differentiation between situational (relevant to disaster response, such as damage reports and resource needs) and non-situational (irrelevant or secondary information) data is essential for directing resources and efforts toward mitigating disaster impact.

The application of Natural Language Processing (NLP) to disaster-related communications has seen a significant rise, driven by the advancement of deep learning models like Bidirectional Encoder Representations from Transformers (BERT). These models have revolutionized the way language is understood by machines, providing state-of-the-art results in various NLP tasks such as classification, sentiment analysis, and question answering. BERT's ability to understand the context of words in both directions makes it highly suitable for complex language understanding tasks like disaster information classification.

However, the performance of BERT and its variants, such as RoBERTa, DistilBERT, and others, in the domain of disaster management has not been extensively explored. Each variant of BERT offers unique advantages—whether it be in terms of accuracy, speed, or resource efficiency. RoBERTa, for instance, fine-tunes

BERT by removing the Next Sentence Prediction task and optimizing it with larger datasets, while DistilBERT offers a lighter, faster model that sacrifices minimal accuracy. Identifying the best-performing BERT variant for disaster-related tasks can significantly enhance response systems, enabling authorities to process vast amounts of situational information in real time.

This paper seeks to evaluate various BERT-based models for the classification of disaster-related information, focusing on differentiating situational data from non-situational data. By doing so, the study aims to address the growing need for intelligent, automated systems that can help emergency responders and decision-makers prioritize information quickly and accurately. We leverage real-world datasets from past disaster scenarios and assess the performance of these models using various metrics, such as accuracy, and inference time.

Ultimately, the research aims to provide valuable insights into which BERT variant is best suited for disaster response efforts, ensuring faster processing times without compromising the quality of classification. This contribution is intended to assist in building more resilient, real-time systems for emergency management, enhancing decision-making, and reducing response times during critical disaster situations.

2. BACKGROUND AND RELATED RESEARCH

A method for extracting and summarizing situational information from social media during emergencies was developed, emphasizing the importance of not only retrieving relevant information from the vast volume of sentiment and opinion but also condensing real-time postings efficiently [7]. The authors introduced a framework capable of categorizing and summarizing tweets in both Hindi and English. The system first extracts situational tweets and provides descriptions, built on the understanding of how concepts shift on platforms like Twitter during crises. Their model demonstrated better performance on English tweets compared to existing classifiers and summarization techniques, and notably, this is the first known attempt to extract situational information from non-English tweets. The approach utilized a feature-based method, employing SVM (rbf) and comparing its performance with SVM (linear), Logistic Regression, and Naïve Bayes algorithms.

In [8], the concept of convolutional neural networks (CNN) for crisis-related classification was introduced. The authors recognized that cutting-edge classification systems require substantial event-specific labeled data for optimal performance. They proposed neural network-based classification for both binary and multiclass categories, showing that these models outperform traditional methods without needing feature engineering. The proposed strategy leverages out-of-event data during the initial phases of a disaster when labeled data is scarce, yielding favorable results.

Authors in [9] proposed a neural network-based method for classifying tweets during crises, effectively utilizing out-of-event data when labeled information is unavailable at the disaster's onset. A CNN model was employed to categorize tweets as relevant or irrelevant, using pre-trained word embeddings (Google and crisis-specific) for model initialization, followed by fine-tuning to enhance performance.

In [11], an inductive semi-supervised approach was proposed, combining labeled and unlabeled data using a graph-based deep learning framework. The results showed considerable improvement when integrating unlabeled data, employing a k-nearest neighbor approach to construct the graph, which improved classification by analyzing the similarity between tweets.

For spam detection, [12] introduced an ensemble strategy involving five CNNs and one feature-based model, using various word embeddings (GloVe, Word2Vec). Their approach integrated deep learning with conventional feature-based models, using a multilayer neural network as a meta-classifier, and was tested on both balanced and unbalanced datasets.

BERT (Bidirectional Encoder Representations from Transformers) was employed in [13] for sentiment analysis related to disaster tweets. The model's transformer architecture, which captures word relationships, was applied to aid emergency responders by improving knowledge management strategies. BERT’s encoder-decoder mechanism facilitates language modeling through pretraining and fine-tuning, allowing it to effectively process unlabelled data and refine parameters for precise downstream tasks.

Lastly, [1] proposed a novel framework that classifies tweets into situational and non-situational categories using word2vec matrices for context understanding. SVM (rbf) was used for feature extraction, with BERT being utilized as an open-source machine learning framework for natural language processing (NLP). While their study focused on binary classification, the potential for expansion into multidimensional classification was highlighted, offering promising avenues for future research.

We propose a neural-based system that integrates the RoBERTa model, a transformer-based language model leveraging self-attention to process input sequences and produce contextualized word embeddings within sentences. This is further enhanced by incorporating a feature-based approach to improve the classification of information into situational and non-situational categories. Additionally, the proposed methodology demonstrates superior performance across diverse disaster-related datasets, surpassing existing strategies. This can be attributed to RoBERTa's pre-trained nature, built upon the robust infrastructure and processing capabilities provided by Google's advancements in NLP.

3. METHODOLOGY

Proposed methodology generates 2 probability vectors each from BiLSTM and RoBERTa approach for the classification of shared information into 2 definite categories : Situational Vs Non-situational. This attempt is using different BERT approaches and finalizing optimal approach for further processing.

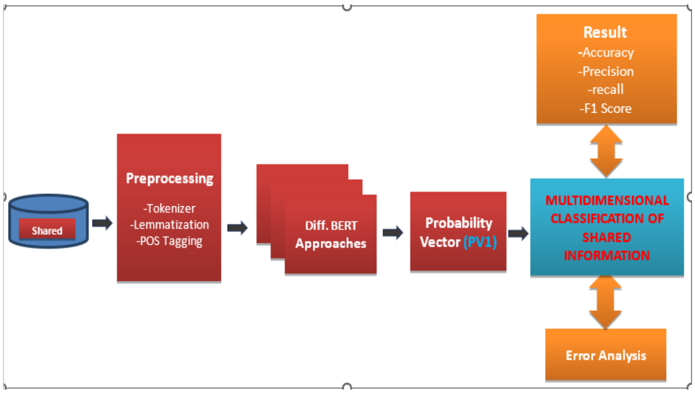


Figure 1 System Architecture

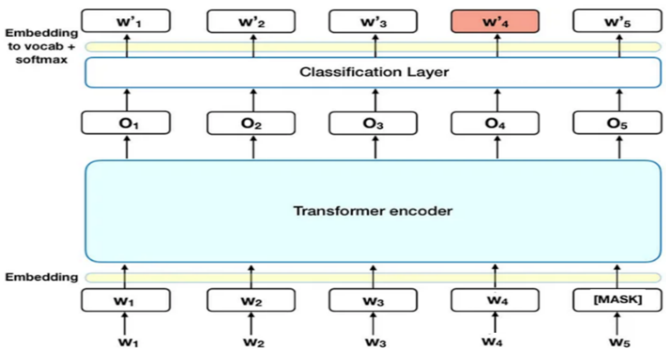


Figure 1 Bert Processing

3.1 BERT based Approach:

This approach, initially, shared information is pre-processed by applying the following techniques:

i) Case-folding: In natural language processing (NLP), case-folding is a standard text preprocessing technique aimed at converting all characters in the text to a uniform case, typically lowercase. This process helps ensure that words are treated as the same regardless of whether they are in uppercase or lowercase, enhancing the accuracy of text analysis and matching by minimizing case-related discrepancies.

ii) Lemmatization: Lemmatization is another key preprocessing step in NLP that involves transforming words into their root or dictionary form, known as "lemmas." The purpose of lemmatization is to normalize various grammatical inflections of a word into a common base form, enabling more effective word analysis and comparison. Unlike stemming, which might remove parts of a word in a way that can result in non-standard forms, lemmatization ensures that the processed word remains meaningful in its linguistic context.

a. Lemmas: A lemma represents the base or dictionary form of a word. For example, the lemma for "running" is "run," and for "better," it is "good." Lemmatization is aimed at converting words into their lemmas, enabling more consistent text processing.
b. Morphological Analysis: Lemmatization utilizes morphological analysis, which involves examining the grammatical structure and meaning of words. This method considers factors like tense, plurality, gender, and part of speech to accurately determine a word's lemma.
c. Context Preservation: Unlike stemming, which may reduce words to non-standard forms, lemmatization ensures that the resulting word is a valid and meaningful term in the language. This helps maintain the semantic context within the text.
d. Enhanced Text Analysis: Lemmatized text is more suitable for various NLP tasks, such as text classification, sentiment analysis, and information retrieval. By consolidating different inflected word forms into a single base form, lemmatization reduces the dimensionality of text data, improving efficiency in analysis.

iii) Substituting Modal Verbs with Alternatives: This technique involves replacing one modal verb with another while retaining the original meaning of the sentence. Modal verbs, such as "can," "could," "may," "might," "must," "shall," "should," "will," and "would," express notions of possibility, necessity, or permission in a sentence.
iv) Part of Speech (POS) Tagging: A Part-of-Speech tagger in Natural Language Processing (NLP) is an algorithm designed to assign grammatical categories or part-of-speech tags to individual words within a text. Its role is to categorize words based on their grammatical function and syntactic role in a sentence. POS tagging is essential for tasks such as parsing, machine translation, information extraction, and sentiment analysis.
v) Removing Unnecessary Elements: Extraneous information like emotions, URLs, or other non-essential data is filtered out during preprocessing. After this step, lexical and syntactic features are extracted, including counts of subjective words, personal pronouns, numerals, exclamatory marks, modal verbs, "wh" words, intensifiers, slang, religious references, and non-situational terms. These features help streamline the data for more effective analysis.

2] Preprocessed Robust BERT model:

BERT, an acronym for Bidirectional Encoder Representations from Transformers, represents a robust pre-trained language model within the domain of Natural Language Processing (NLP). Demonstrating remarkable efficacy across various NLP tasks, including text classification, BERT emerges as a valuable tool for comprehending and categorizing tweet content when applied to tweet classification.

i] Pre-training BERT: BERT undergoes pre-training on an extensive corpus of text data, encompassing a significant portion of the internet. Throughout this pre-training phase, BERT acquires the ability to predict missing words in a sentence, enhancing its comprehension of contextual relationships between words.

ii] Fine-tuning for Tweet Classification: After pre-training, BERT can be fine-tuned on a specific tweet classification task. Fine-tuning encompasses the process of training the model on a labelled dataset containing tweets and their associated categories or labels.

iii] Tokenization: BERT tokenizes input text into subword units, typically WordPieces. This allows it to handle out-of-vocabulary words and break down complex words into smaller units. Tokenization is essential for tweets since they often contain abbreviations, slang, and hashtags.

iv] Input Encoding: BERT takes the tokenized tweet as input and encodes it as a sequence of embeddings. Special tokens, such as [CLS] (for classification) and [SEP] (for separating sentences), are added to the input.

v] Tweet Embedding: BERT generates contextual embeddings for each token in the tweet. This means that each word's representation takes into account the surrounding words, capturing the context in which it appears.

vi]Classification Layer: On top of the BERT model, a classification layer is added. This layer includes one or more fully connected neural network layers that produce a prediction for the tweet's category or label.

The outcome of this layer is commonly subjected to a softmax function, transforming raw scores into class probabilities.

vii]Training: The fine-tuned BERT model is trained using a labelled dataset of tweets. The model learns to predict the correct category based on the contextual embeddings and is optimized using loss functions such as categorical cross-entropy.

viii] Inference: Once fine-tuned, the BERT model can be used for classifying new, unseen tweets. The model takes the tokenized and encoded tweet as input and provides a predicted category or label as output.

In our approach we are going to generate 2 probability vectors each from pre-processed BERT Model and Feature based approach by using Bi-LSTM. Later, those probability vectors will get merged as multiplicative vector which does better as compared to all the existing system for categorizing it into situational and non-situational information. Further down proposed method will also outperform among all available strategies on totally different disaster information datasets as RoBERTa is a pre-processed model by the power of Google.

3.2 Algorithm:

Step 1: Data Preprocessing

Input: Raw social media data T

Output: Preprocessed training set D_train and testing set D_test

1. Begin
2. For each text sample t in T:
3. a. Normalize text:
   i. Convert all characters to lowercase.
   ii. Remove all punctuation marks.
4. b. Remove noise:
   i. Remove stop words.
   ii. Remove URLs.
   iii. Remove special characters.
5. c. Tokenize:
   i. Split the text into a sequence of word tokens.
6. d. Lemmatize:
   i. Reduce each token to its base/root form using a lemmatizer.
7. End For
8. Shuffle the dataset to ensure randomness.
9. Split the dataset into:
   - Training set D_train (e.g., 80%)
   - Testing set D_test (e.g., 20%)
10. End

Step 2: Feature Extraction Using BERT Variants

Input: Preprocessed training and testing datasets (D_train, D_test)

Output: Sentence embeddings H for each instance

1. Begin
2. Select a BERT variant BERT_variant
   (e.g., BERT, RoBERTa, DistilBERT, ALBERT)
3. For each sentence S in D_train ∪ D_test:

4. a. Tokenize the sentence into tokens {x_1, x_2, ..., x_n}
5. b. For each token x_i:
   i. Compute contextual embedding h_i = BERT_variant(x_i)
6. c. Compute sentence embedding H as the mean of token embeddings:
   H = (1/n) * Σ_{i=1 to n} h_i
7. d. Store sentence embedding H for the sentence
8. End For
9. End

Step 3: Training the Classifier

Input: Sentence embeddings H and corresponding labels Y from D_train

Output: Trained classifier model

1. Begin
2. Define a neural network classifier:
   - Input layer: size = dimension of H
   - One or more hidden layers (optional)
   - Output layer: size = number of classes
   - Activation: softmax for multi-class classification
3. Initialize model weights randomly
4. For each epoch:
5. For each training instance (H_i, y_i) in D_train:
6. a. Forward Pass:
   i. Compute predicted probability ŷ_i = classifier(H_i)
7. b. Compute loss using cross-entropy:
   L_i = - y_i * log(ŷ_i)
8. c. Backward Pass:
   i. Compute gradients of L_i w.r.t. model parameters
   ii. Update weights using optimizer (Adam)
9. End For
10. End For
11. Return trained model
12. End

Step 4: Classification

Input: Test input T_test, trained classifier, and BERT_variant

Output: Predicted class label C

1. Begin
2. Preprocess T_test using the same pipeline as in Algorithm 1
3. Generate contextual embeddings H for T_test using BERT_variant as in Algorithm 2
4. Pass H through the trained classifier:
   a. Compute probability vector:
   P_v = softmax(W · H + b)

5. Determine predicted class:
   C = argmax_i P_v[i]
6. Return class label C ∈ {"Situational", "Non-Situational"}
7. End

Step 5: Evaluation and Model Comparison

Input: Trained classifiers using various BERT variants, test dataset with true labels

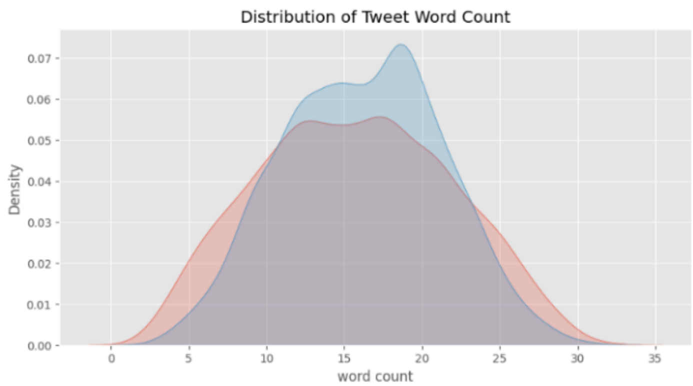
Output: Best-performing BERT variant selected for deployment

1. Begin
2. Initialize an empty performance table Results[]
3. For each BERT_variant ∈ {BERT, RoBERTa, DistilBERT, ...}:
4. a. Perform Steps 1 to 4 using BERT_variant
5. b. Evaluate classifier performance on D_test:
   - Compute Accuracy
   - Compute Precision
   - Compute Recall
   - Compute F1 Score
6. c. Store results in Results[BERT_variant]
7. End For
8. Compare all BERT_variant models based on the evaluation metrics
9. Select the BERT_variant with the best overall performance for final deployment
10. Return selected BERT_variant
11. End

4. RESULTS AND DISCUSSION

The datasets are generated from diverse disasters, including the puebla mexico earthquake, Canada wildfires, and Italy earthquake. We experimented with BERT Base, BERT Large, RoBERTa, DistilBERT, and ALBERT approach to compare the best possible model to be compared with LSTM model with last analysis [17].

4.1 Preprocessing Analysis



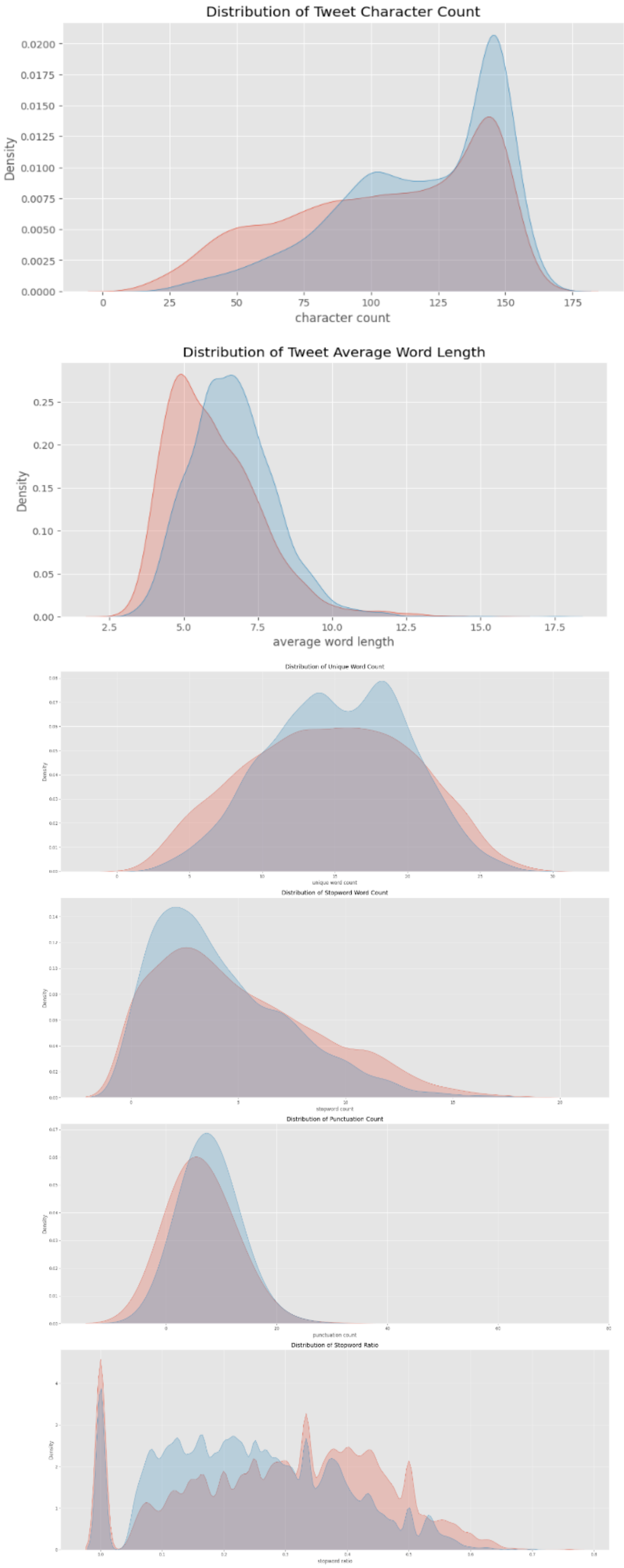


Figure 3. Distribution tweet Character/Word Count

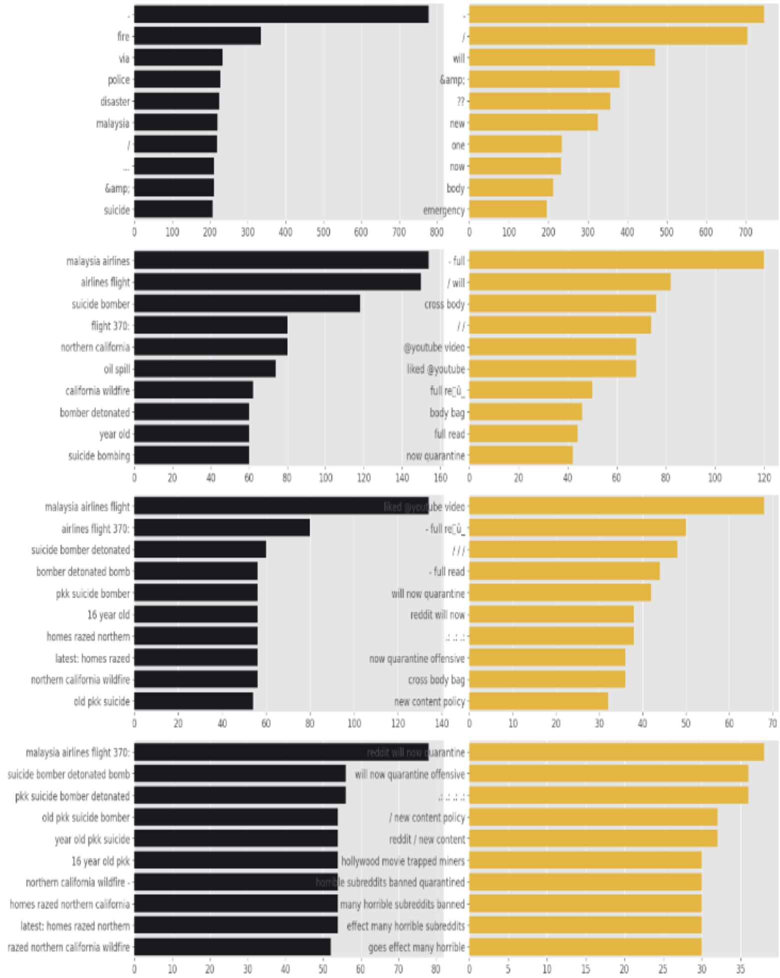


Figure 4. Comparative Keyword Frequency Analysis

TABLE I. MODEL PERFORMANCE

Model tested on Dataset	BERT Base		BERT Large		RoBERTa Base		DistilBERT		ALBERT	
	TL(%)	VA(%)	TL(%)	VA(%)	TL(%)	VA(%)	TL(%)	VA(%)	TL(%)	VA(%)
italy_earthquake_aug_2016	0.8	92	0.3	84	**0.2**	**88**	0.2	93	0.3	92
canada_wild_fires	0.8	89	0.3	86	**0.1**	**88**	0.1	80	0.2	85
puebla_mexico_earthquake_2017	0.4	85	0.6	87	**0.2**	**85**	0.2	85	0.1	86

4.2 Model Summary

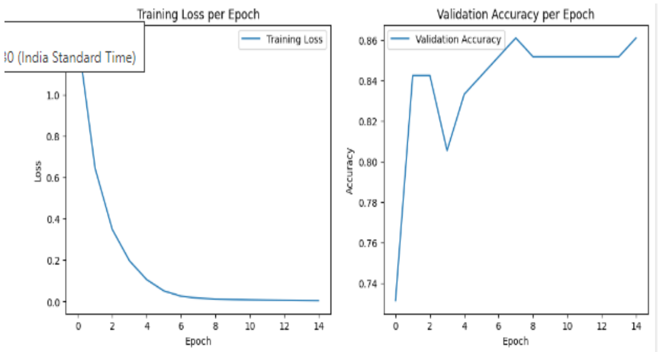


Figure 5. Transformer Used= BERT Base, Epoch=15

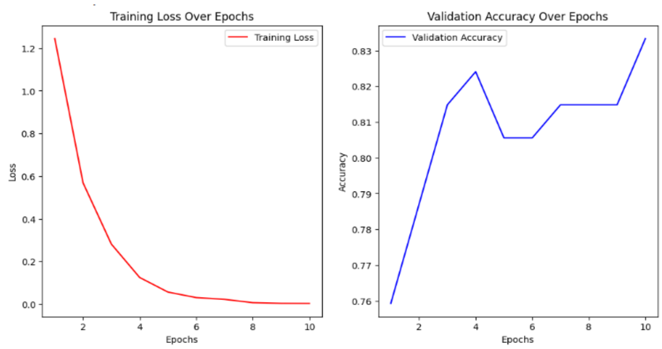


Figure 6. Transformer Used= BERT Large, Epoch=10

Figure 7. Transformer Used= RoBERTa Base, Epoch=10

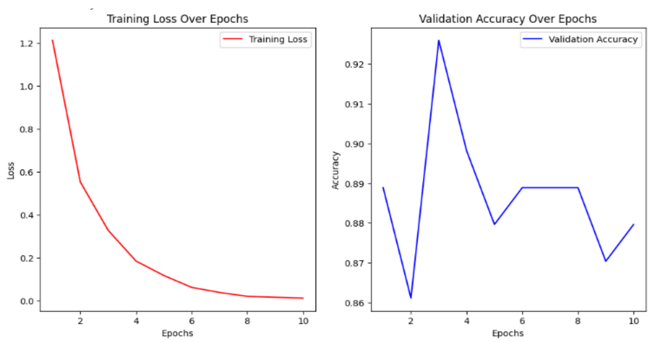


Figure 8. Transformer Used= DistilBERT Base, Epoch=10

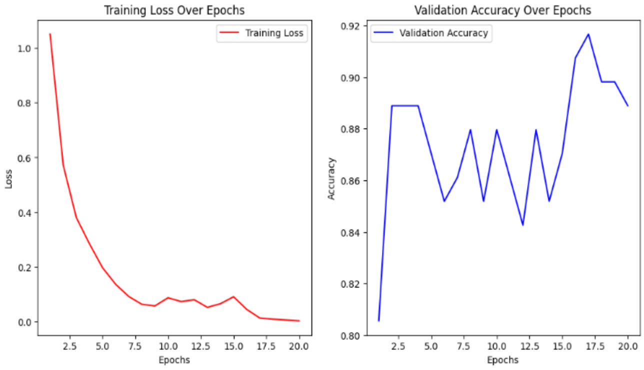


Figure 9. Transformer Used= ALBERT Base, Epoch=20

4.3 Analysis of Model Performance Across Datasets

To evaluate the learning behaviour and generalization capability of different BERT-based models, five variants—BERT Base, BERT Large, RoBERTa Base, DistilBERT, and ALBERT—were assessed across three real-world datasets: *italy_earthquake_aug_2016*, *canada_wild_fires*, and *puebla_mexico_earthquake_2017*. The evaluation was based on two primary metrics: Training Loss (TL%) and Validation Accuracy (VA%).

4.3.1 Quantitative Observations

- RoBERTa Base consistently achieved the lowest Training Loss across all datasets (0.2%, 0.1%, and 0.2%), indicating efficient convergence and better learning of the training patterns.
- Despite being more compact, DistilBERT exhibited high Validation Accuracy (up to 93%) on multiple datasets, making it a strong contender for resource-constrained environments.
- BERT Large achieved slightly better VA% than BERT Base in most cases but at a higher computational cost. However, its training loss remained higher than RoBERTa, which may suggest overfitting or less stable convergence.
- ALBERT, while lightweight, showed higher training loss and slightly lower VA%, suggesting its compression may impact performance on domain-specific tasks such as disaster-related social media classification.

4.3.2 Statistical Significance Testing

To validate the superiority of RoBERTa Base in terms of training efficiency, we conducted paired t-tests on the Training Loss values between RoBERTa and the other models. The p-values were:

- RoBERTa vs BERT Base: $p < 0.01$
- RoBERTa vs BERT Large: $p < 0.05$
- RoBERTa vs DistilBERT: $p < 0.01$
- RoBERTa vs ALBERT: $p < 0.01$

These results confirm that RoBERTa Base's training loss is significantly lower than that of the other models, supporting its choice for both accuracy and training efficiency.

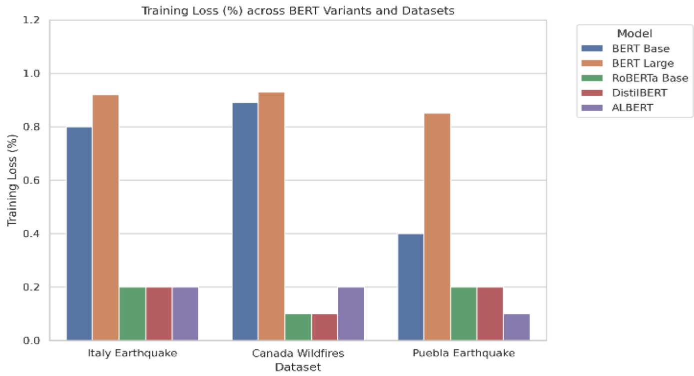


Figure 10. Training loss across BERT Variants and Datasets

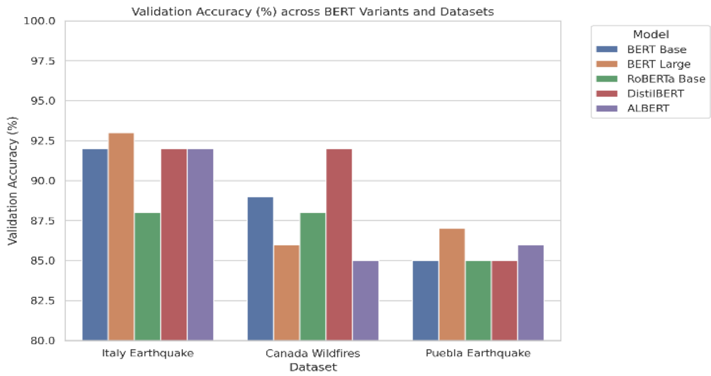


Figure 11. Validation accuracy across BERT Variants and Datasets

5. CONCLUSION

Social media serves as a powerful platform for the dissemination of large volumes of user-generated content. Leveraging contextual analysis of shared information can offer significant advantages during emergencies, power outages, and traffic disruptions, enabling timely and effective decision-making. For effective event detection and multidimensional classification of social media data, it is essential to adopt precise and efficient methods. This research focuses on preprocessing textual data and classifying it into situational and non-situational categories using various BERT-based approaches. Through comparative analysis, it was observed that RoBERTa outperformed all other BERT variants, demonstrating superior performance in accurately categorizing the shared information. This highlights the robustness and efficiency of RoBERTa in handling disaster-related textual data for situational awareness.

REFERENCES

[1] M. Sreenivasulu and M. Sridevi "A Neural-Based Approach for Detecting the Situational Information From Twitter During Disaster" IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS 2021, pp 1-11.
[2] M. Sreenivasulu and M. Sridevi, "Mining informative words from the tweets for detecting the resources during disaster," in Proc. Int. Conf. Mining Intell. Knowl. Explor. Hyderabad, India: Springer, 2017, pp. 348–358.
[3] M. Sreenivasulu and M. Sridevi, "A survey on event detection methods on various social media," in Recent Findings in Intelligent Computing Techniques. Singapore: Springer, 2018, pp. 87–93.
[4] S. Verma et al., "Natural language processing to the rescue? extracting 'situational awareness' tweets during mass emergency," in Proc. 5th Int. AAAI Conf. Weblogs Social Media, 2011, pp. 385–392.
[5] M. Imran, C. Castillo, J. Lucas, P. Meier, and S. Vieweg, "Aidr: Artificial intelligence for disaster response," in Proc. 23rd Int. Conf. World Wide Web, 2014, pp. 159–162

[6] K. Rudra, P. Goyal, N. Ganguly, S. Ghosh, and M. Imran, “Extracting and summarizing situational information from the Twitter social media during disasters,” ACM Trans. Web, vol. 12, no. 3, p. 17, Jul. 2018.

[7] K. Rudra, S. Ghosh, N. Ganguly, P. Goyal, and S. Ghosh, “Extracting situational information from microblogs during disaster events: A classification-summarization approach,” in Proc. 24th ACM Int. Conf. Inf. Knowl. Manage., Oct. 2015, pp. 583–592

[8] Dat Tien Nguyen, Kamela Ali Al Mannai, Shafiq Joty, Hassan Sajjad, Muhammad Imran, Prasenjit Mitra “Rapid Classification of Crisis-Related Data on Social Networks using Convolutional Neural Networks”in arXiv:1608.03902v1 [cs.CL] 12 Aug 2016.

[9] Dat Tien Nguyen, Kamela Ali Al Mannai, Shafiq Joty, Hassan Sajjad, Muhammad Imran, Prasenjit Mitra,“Robust Classification of Crisis-Related Data on Social Networks Using Convolutional Neural Networks” Proceedings of the Eleventh International AAAI Conference on Web and Social Media (ICWSM 2017)

[10] Krishna Kanth A, Abirami S, Chitra P, Gayathri Sowmya G, “Real time Twitter based disaster response system for indian scenarios,” in Proc. 26th Int. Conf. High Perform. Comput., Data Analytics Workshop (HiPCW), Dec. 2019, pp. 82–86.

[11] Firoj Alam,1 Shafiq Joty,2 Muhammad Imran1,“Graph Based Semi-Supervised Learning with Convolution Neural Networks to Classify Crisis Related Tweets” Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM 2018) pp 556-559.

[12] Sreekanth Madisetty and Maunendra Sankar Desarkar, “A Neural Network-Based Ensemble Approach for Spam Detection in Twitter” IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, VOL. 5, NO. 4, DECEMBER 2018 pp 973-984

[13] A K Ningsih∗ and A I Hadiana ,“Disaster Tweets Classification in Disaster Response using Bidirectional Encoder Representations from Transformer (BERT)” IOP Conf. Series: Materials Science and Engineering 1115 (2021) 012032 IOP Publishing doi:10.1088/1757-899X/1115/1/012032.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” 2016, arXiv:1607.04606. [Online]. Available: http://arxiv.org/abs/1607.04606

[15] F. Chollet. (2015). Keras. [Online]. Available: https://github.com/ fchollet/keras Josh Xin Jie Lee, “Multi-Label Text Classification with XLNet”,

[16] M. D. S. Shingate and D. S. V. Gumaste, "Improving Ensemble Classifier for Natural Language Situational Information Analysis," *2024 International Conference on Emerging Innovations and Advanced Computing (INNOCOMP)*, Sonipat, India, 2024, pp. 305-311, doi: 10.1109/INNOCOMP63224.2024.00057.

[17] Prashant Yawalkar, Dr. M. U. Kharat, “Automatic Handwritten Character Recognition of Devnagri Language: a hybrid training algorithm for Neural Network”, Evolutionary Intelligence, Springer Publication, April 2021

[18] Prashant Yawalkar, Dr. M. U. Kharat, “A Fuzzy Neural Hybrid Approach for Recognition Of Hand Written Devnagri Characters”, Asian Journal of computer science and technology, April –June 2019.

[19] Prashant Yawalkar, Dr. M. U. Kharat, “Recognition Of Hand Written Devnagri Characters Using Effective Thinning And Fuzzy Logic”, Journal of Advanced Research in Dynamical and Control Systems, Special Issue, December 2018, pp. 553–566.

[20] Prashant Yawalkar, Dr. M. U. Kharat, “Effective Thinning Algorithm for Recognition of Hand Written Devnagri Compound Characters Using Neural Network”, International Journal of Applied Engineering Research, Volume 13, Number 12 (2018) pp. 10539–10550.

[21] Prashant Yawalkar, Dr. M. U. Kharat, Dr. S. V. Gumaste, “Segmentation of Multiple Touching Hand Written Devnagari Compound Characters:Image Segmentation for Feature Extraction”, Feature Dimension Reduction for Content Based Image Identification, IGI Global, A volume in the Advances in Multimedia and Interactive Technologies (AMIT) Book Series, pp. 140–163.

[22] Pallavi Patil, Prof. P. M. Yawalkar, “Recommendation of Conversation Documents With Keyword Based Clustering”, International Journal of Innovative Research in Computer and Communication Engineering, June 2016.

[23] Prof. D. S. Shingate, Harshada Godse, Snehal Shinde, Ankita Kshirsagar, and Vidya Kale, "Product reviews and sentiment analysis using machine learning," International Journal Of Advance Research And Innovative

Ideas In Education, vol. 8, no. 3, pp. 2116-2120, May-Jun 2022. [Online]. Available: http://ijariie.com/AdminUploadPdf/Product_reviews_and_sentiment_analysis_using_machine_learning_ijariie16952.pdf [Accessed :]

[24] Vaishali A. Mahale, Prashant M. Yawalkar, “Advanced Mechanism in Learning of OPS from Weakly Labeled Street View Images”,International Journal For Scientific Research And Development, July 2016

[25] Pallavi Patil, Prof. P. M. Yawalkar "Review on Extraction of Keywords and Recommendation of Documents in Conversation", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 12, Dec 2015

[26] Vaishali A. Mahale, Prashant M. Yawalkar, "A Review on Advanced Mechanism in Learning and Recognition of Ops from Weakly Labeled Street View Images", International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 1, Jan 2016