Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Sentiment Analysis for Disaster Management on Tweeter Data: Machine Learning and Deep Learning Approaches

Soniya Zore, Amol Bhosale, Dr..Pratibha Chavan, Anil Sawant, Radhika Bodhe

Department of Electronics and Telecommunication, Trinity College of Engineering & Research, Pune, India
Email: soniyazore93@gmail.com

Disasters can strike suddenly, causing widespread devastation and requiring a swift response. Social media, particularly Twitter, has become a valuable platform for sharing information during emergencies. Analyzing this real-time data can aid disaster relief efforts by various ways. Assessing public sentiment is one of the effective techniques amongst them. The proposed work focuses on sentiment analysis on twitter data using comparative analysis of various ML DL algorithms. Experimentation show domination of Naive Bayes technique (80% accuracy) for positive or negative sentiment recognition from live or real time input tweet. The Naive Bayes classifier outperformed on others viz. Random Forest, SVM and LSTM techniques. The machine learning model developed by us can be expanded by incorporating comments on other social media platforms. The use of some adaptive algorithms may bring additional efficiency to the system.
Keywords: SVM, LSTM, Twitter.

1. Introduction

Wide use of social media and change in human behavior towards reacting or commenting instantly on any situation is a paradigm shift in today's world. Such comments make positive as well as negative impact suddenly causing widespread devastation and requiring a swift response. Twitter is one of the trendiest social media platforms for sharing information during emergencies. The Lakh of comments collecting on this platform can be analyzed on real-time basis to aid disaster relief. The usage of analysis methods is tabulated in table 1.

Sr. No.	Usage	Parameters / Attributes
1	Identifying affected areas	Geotagged Tweets (GPS) Location Mentions (NLP) Spatial Clustering Image and Video Analysis
2	Understanding nature of disaster	Content (keywords, location) Severity (Urgency, desperation)
3	Assessing public sentiments and needs	Sentiments (Fear, frustration, hope etc.) Keyword (Food, medical, shelter etc.) Hashtags (Specific needs) Call to action tweets

Table 1- Usage of comment analysis methods

Sentiment analysis, a field of natural language processing (NLP), has come out as a crucial tool for understanding public opinion, emotional responses, and attitudes expressed on social media platforms like Twitter. With the exponential growth of user-generated content on Twitter, ranging from personal anecdotes to global events, sentiment analysis offers valuable insights into the collective mood of the platform's diverse user base.

This burgeoning field utilizes advanced algorithms and machine learning techniques to categorize tweets as positive, negative, or neutral based on the underlying sentiment conveyed in the text. By analyzing linguistic cues, context, and emotive expressions, sentiment analysis enables researchers, businesses, and policymakers to gauge public sentiment on various topics, ranging from product reviews and brand perceptions to social issues and political discourse.

In this work, we delve into the realm of sentiment analysis on Twitter, exploring positive or negative comments based on real-time input tweet. The work flow in below sections elaborates our contribution. State-of-art methodologies suggested by various researchers are given in section II. The raw data converted to usable form by pre-processing explained in section III. The details of ML DL algorithms used for sentiment analysis are given in section IV. Experimentation and outcomes are elaborated in section V, after that conclusion and future scope explored in section VI.

2. Literature Review

Several researchers have proposed strategies for classifying tweets based on their informative contents. One approach involves leveraging deep learning techniques, which have demonstrated significant success in both image and text processing domains. For instance, in the realm of image processing, there are instances where articles contain images and titles that are not directly related.

To embark upon this issue, paper [1] introduced a convolution neural network (CNN) model. This model predicts similarities using image and text features between them. By extracting features from both modalities, the model offers insights into the association between images and text, potentially enhancing search quality.

The utmost spoken language across world is English. Most researchers preferred using English as a language for sentiment analysis. A. Tumasjan et.al., did Twitter based prediction in 2009 and found that half population of users (50%) reacted only one time in their corpus on twitter against 90% portion of tweets covered by rest 50% tweets [2].

A big dataset of approx. 100K tweets is really indefinite to predict country's large population. B. Joyce and J. Deng congregated tweets from the presidential election of 2016 highlighting majorly used specific words viz. Hillary Clinton and Donald Trump [3]. Use of Twitter's API helped them for such huge quantity of tweets collections. Words like 'democratic' and 'republican' are observed during this collection along with full name of the candidates in race viz. Donald Trump. Authors collected around 79 million tweets marked over a span of two months via 10,000 unique users. Around 2.4% tweets out of them are filtered bringing 1.9 million tweets containing emojis out of table. The redundancies caused due to repeated tweets are eliminated further to deplete the dataset to 783K tweets. Using the sentiment ranking of emoji, they calculated each tweet's emotional direction. For that, emoji extracted from the sentence are converted into Unicode. Then applied regex on Unicode and originated a score from 522 unique emoji characters in the customize list. This score used to calculate emotional direction or sentiment of the sentence. A multinomial Naïve Bayes classifier is suggested to build a sentiment classifier, in this work. To test the classifier, they used hand-annotated tweets as test data and passed it to the classifier. They achieved 74.9 % accuracy on this test data. Considering the fact of non-occurrence of emoji in each sentence, it should not be treated as only parameter to judge the result. The use of emoji as a sarcastic tone needs to be taken into account. In addition to this, word ordering plays a sizable role in sentiment analysis. For example, one sentence is "I have to go to travel by car" and another sentence is "I have car to travel". Both examples have different meaning. Many such examples can justify the weightage to be given for word ordering.

Additionally, the evolving tactics employed by spammers necessitate robust solutions that can circumvent traditional security mechanisms. Paper [4] proposed a method that employs low-level n-gram features to thwart tokenization, thereby addressing this challenge. Researchers have explored various approaches for analysing language-dependent devices, utilizing publicly available databases to assess the performance of multiple systems. They incorporate n-gram analysis from tweet tones into development methodologies. Notably, they have demonstrated the capability of technology to swiftly detect spam tweets, a critical aspect in real-time Twitter scenarios. Twitter's significance as a data source has propelled the reputation of tweet sentiment analysis.

In this context, paper [5] introduces a method for the limited analysis of typical tweets by adopting a classification strategy with notable successes. Given the prevalence of emotionally charged phrases in tweets, effectively addressing specific conceptual phrases could significantly enhance cognitive analysis techniques' efficacy. The study ascertains a methodology focused exclusively on analysing tweets containing positive emotion phrases, showcasing promising results across both tasks.

The challenges inherent in social media data analysis, such as noise, brevity, and the need to categorize incoming messages, underscore the importance of access to human-specific information. To this end, paper [6] employs machine learning classifiers trained on adjective usage, alongside releasing a word2vec software trained on a vast corpus of disaster-related tweets. Additionally, the author addresses language variations in tweets by providing common lexical resources for lexical variants.

In the realm of tweet informativeness and event detection, paper [7] proposes a Convolutional Neural Network (CNN)-based method. This approach involves training a CNN model on recent earthquake-related tweets, with a focus on discerning informative tweets and real-time event detection. The CNN model plays a pivotal role in predicting Twitter keywords associated with seismic events, aiding in post-event earthquake detection with a high level of accuracy and pre-announcement confirmation from official disaster sources. Notably, existing programs in the literature primarily rely on small datasets or necessitate logical variable names for operation.

While machine learning models surpass lexicon and rule-based algorithms in sentiment classification tasks, in paper [8] LSTM model's accuracy doesn't meet expectations. However, there are avenues for enhancing its performance. For instance, the use of GloVe for word embedding lacks emoticon vector representations, hindering the model's ability to interpret emoticons in text. Employing alternative embedding with emoticon vectorizations could improve accuracy. In retrospect, opting for different embedding and integrating attention mechanisms to prioritize emoticons and punctuations would be strategies to boost the model's effectiveness.

Pradip Bhereet. al., [9] introduced a tweet classifier framework designed to process raw tweets and classify them as informative or non-informative. The model has demonstrated superior performance in comparison with state-of-art system that employed a mixture of CNN and ANN. By leveraging both the Continuous Bag of Words (CBOW) and Skip-Gram models of Word2Vec in conjunction with Convolutional Neural Networks (CNNs), achieved an assessment accuracy of 84%. Deep learning methods tested by various researches are studied, compared and overviewed in detail in [10]. The review is majorly presented in categories viz. algorithmic developments with data concepts, knowledge concepts and text mining. Another survey in [11] showcases various proposals for sentiment analysis and highlight major challenges in the field. An innovative model named as BERT is put forth in [12] produces absolute improvement in different performance metrics and outperforms human performance. The application oriented review of sentiment analysis procedures is taken by Jyoti Yadav in 2023 summarizes challenges with respect to each application [13]. Xiliang Zhu et.al., tackled the need of models to deal with diverse linguistic data [14]. The integration of various modalities like text, video or image is done in [15] and out warded richer sentiment insights. The evaluation of sentiment analysis algorithm on real time basis is proposed with a model processing the huge amount of data in [16]. Authors used data ingestion tools namely twitter API and Apache Flume which produced better performance than state-of-art methods. Another similar work obtaining 90% accuracy aggregated twitter API, Apache Kafka and Apache Spark for live sentiment prediction. They explored and visualized the data using modern tools like Elasticsearch and Kibana [17]. V Kharde and S Sonawane demonstrated how sentiment analysis can be used in opinion mining [18]. Sudhir Sharma et.al, explored how companies track brand sentiment and respond to customer feedback [19].

In conclusion, the literature surveyed underscores the multifaceted challenges and advancements in the classification and analysis of tweets across various domains. Deep learning computation, especially convolutional neural networks (CNNs), have shown guarantee in extracting meaningful insights from both textual and visual content, thereby enhancing search quality and predictive capabilities. Additionally, the relentless evolution of spamming tactics necessitates robust solutions, with low-level n-gram features emerging as a viable approach to thwart tokenization and improve detection accuracy.

Furthermore, sentiment analysis on Twitter has emerged as a vital tool for gauging public opinion and understanding emotional responses to diverse topics. Leveraging machine learning classifiers and word embedding trained on vast tweet datasets, researchers have made strides in accurately categorizing tweets based on their sentiment. However, challenges persist, particularly in interpreting emoticons and punctuations, highlighting the need for innovative approaches such as attention mechanisms to enhance model performance.

In summary, while considerable progress has been made in tweet classification and sentiment analysis, there remains ample room for further research and innovation. Continued exploration of advanced techniques and methodologies will be crucial in unlocking the full potential of tweet data for diverse applications ranging from disaster detection to election forecasting and beyond.

To conquer with above issue, we developed a model that estimates the sentiment anchored by word number. Deep learning approach like Long-short Term Memory [LSTM] is used in this work. It is a type of the Recurrent Neural Network techniques. Here model we trained with the word’s ordering instead of consideration of emojis in the past analysis by researchers. By reason of the above parameters, LSTM gives the impression to be the robust for applying sentiment analysis. Thus, we incorporated the use of LSTM model for this analysis.

3. Pre-processing

A Data Acquisition: A publicly available Twitter disaster dataset is used [20]. This dataset should be labelled, indicating whether a tweet is related to a disaster or not. Obtaining Twitter data for disaster analysis requires strategic planning. Here are two main approaches:

1. Twitter API:

- Application: Apply for a developer account with Twitter to access the Twitter API. This grants programmatic access to tweet data.
- Keyword Filtering: Define relevant keywords associated with disasters (e.g., "flood," "earthquake," "hurricane").
- Location Targeting: Specify locations of interest or filter based on geotagged tweets to pinpoint affected areas.
- Historical vs. Real-Time: Choose between gathering historical data for past events or collecting real-time data during ongoing disasters.

2. Streaming Tools:

- Tweepy (Python): Utilize libraries like Tweepy (for Python) to connect to the Twitter Streaming API and collect tweets in real-time.

- Focus on Speed: Ideal for capturing the fast-moving nature of disasters and gathering the latest updates.
- Data Volume: Be prepared to handle a potentially high volume of tweets during a disaster.

Additional Considerations:

- Rate Limits: Twitter API enforces rate limits on data collection requests. Plan your queries accordingly to avoid exceeding limits.
- Data Filtering: Develop a filtering process to eliminate irrelevant tweets (e.g., advertisements, spam) and focus on disaster-related information.
- Ethical Considerations: Respect user privacy and avoid collecting or using personally identifiable information (PII) without proper consent.

By effectively utilizing the Twitter API or streaming tools, we acquired valuable data for disaster analysis. One of most popular data repository named as Kaggle is used to get the dataset. It is renowned online community for data scientist and machine learning (ML) practitioners. We basically tackled the problem labelled "Natural Language Processing with Disaster Tweets", which is a part of a competition launched on the website. The overall dataset contains total 7560 tweets separated in a training dataset (comprising 4983 rows) and testing dataset (encompassing 2567 rows). The two datasets majorly differ on the basis of known target attribute. Due to non-availability of targets in testing dataset and in order to be able to evaluate the model after all, we chosen only training dataset accordingly and used it as the main dataset. Further it is divided in training-testing subsets.

B Data Pre-processing

The raw contents of various posts tweeted by users are hardly utilizable because they are caused by their different characteristics. Thus, a pre-processing on this raw data is must to normalize the text. The major steps we applied by focussing on reduction in feature set size so that it became usable for learning algorithms. These pre-processing steps opted by us are elaborated as given below.

1) Hashtags: A hashtag is a keyword (named as it is prefixed with the hash symbol (#)) or a no-spaced phrase. They get used in naming subjects and phrases and generally get linked with current trending topics, e.g.-, #Election2024, #SpiderManNoWayHome etc.

2) Handles: Each Twitter user gets unique username allotted upon sign-up. Any information or tweet directed towards that user can be indicated by writing their username preceded by '@'. Thus, it gives a feel of proper nouns, e.g.- @ViratK, @rogerfederar etc. Handles are notified by the expression HNDL \1.

3) URLs: Users often used to share hyperlinks along with their tweets. These hyperlinks get shorten with the use of Twitter's in-house URL shortening service. The links like http://t.co/FCWXoUd8 - also facilitate Twitter to alert users if the link founds to be out of domain. From the point of view of sentiment classification, the presence of a URL is most prominent feature than a particular URL. We replaced URLs by the keyword URL.

4) Emoticons: The use of emoticons is highly observed on social media websites. We

identified these emoticons, categorized them in different types and substituted them by one of these suitable keywords listed here viz. EMOT SMILEY, EMOT LAUGH, EMOT LOVE, EMOT WINK, EMOT FROWN, EMOT CRY.

5) Punctuations: In view of sentiment classification, all punctuations may not be significant but some of them, like question mark, full stop, exclamation mark directly depict information concerning the sentiments of the text. In line with this observation, we substituted each word boundary by a list of timely found relevant punctuations.

6) Construct combined dataset: Amongst various labels prominently observed in our dataset, we focussed on the tweets that having annotations such as positive, negative, and neutral. Tweets apart from these annotations are removed by marking as as irrelevant or other. The pre-processing code suggested in [15] is applied for normalization of tweet data. These pre-processed tweets are merged into single dataset including their labels. Random shuffling is done to divide the dataset into training (80%) and testing (20%) groups.

4. Sentiment Prediction Methodology

The prediction of any input data based on training need feature extraction in case of applying machine learning algorithms. The features included in our work are-

- Tweet length: Length of the tweet text.
- Presence of disaster-related keywords: Identify and count keywords commonly associated with disasters (e.g., "flood", "earthquake", "hurricane")
- Exploratory Data Analysis

The purpose of applying ML or DL algorithms is to identify the content related to disaster labelled as ‘yes’ or ‘no’. The most common keywords categorized with these labels are given in fig.1.

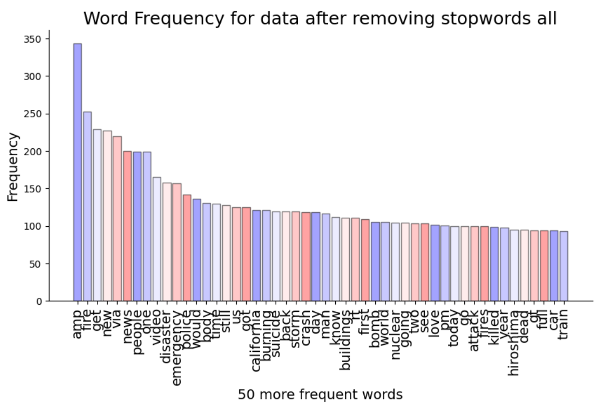

Fig.-1: Common Keywords in disaster contents

Word Embedding-

In addition to use above features for ML algorithm input, word embedding is also used before applying DL algorithm namely LSTM. Since LSTM models work with numerical data, we need to convert the pre-processed text into vectors. Word embedding techniques like Word2Vec or GloVe will be used to represent each word as a numerical vector, capturing its semantic meaning and relationships with other words.

Modelling-

The most popular ML algorithms namely Naive Bayes (NB), Support Vector Machines (SVM), and Random Forest (RF) are tested for our datasets. A well-liked deep learning algorithm titled Long Short-Term Memory (LSTM) is also compared with above three ML algorithms. Steps used in ML and DL techniques are summarized in table 2.

LSTM- Figure 2, depicts a work flow of Long Short-Term Memory (LSTM) network, belonging to recurrent neural network (RNN) architecture preferably suggested for sequential data processing tasks like natural language processing (NLP).

	ML	DL
Pre-Processing	• Tokenization, • Lemmatization • Stemming • Part-of-Speech (POS) tagging • Entity recognition (NER) • Stop word removal	
Feature Extraction and Modelling	One-hot Encoding Bag-of-words representation TF-IDF (Term Frequency-Inverse Document Frequency)	Dense Word Embedding Neural Network with hidden layer
Modelling	NB SVM RF	LSTM

Table 2- Steps used in ML DL algorithm implementation

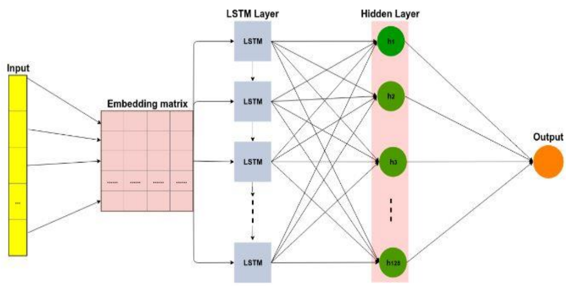

Fig.-2: LSTM work flow

The brief about these steps is summarized as-

1. Input: The input consists of sequential data, such as a sequence of words in a sentence or a time series of data points. In the context of NLP, each input word is typically represented as a numerical vector using techniques like word embeddings.

2. Embedding Matrix: The input is transformed into dense vector representations known as embeddings. These embeddings capture semantic relationships between words and are

become skilled during the training process.

3. LSTM Layer: The LSTM layer consists of multiple LSTM cells stacked on top of each other. Each LSTM cell processes one part of the input sequence at a time and maintains an internal state that captures relevant information from previous elements in the sequence.

The LSTM cells have three main components: the input gate, the forget gate, and the output gate, which regulate the flow of information through the cell and help prevent the vanishing gradient problem commonly encountered in training RNNs.

4. Hidden Layer: The output of the LSTM layer is passed into a hidden layer consisting of multiple neurons or units. Each neuron in the hidden layer receives input from all the LSTM cells in the previous layer. The hidden layer performs nonlinear transformations on the input data, allowing the network to learn complex patterns and relationships.

5. Output: The output layer consists of a single neuron that generates the ultimate outcome of the network. In the context of sentiment analysis or similar tasks, the output neuron typically produces a probability score indicating the predicted class label (e.g., positive or negative sentiment). The output is computed based on the activations of the neurons in the hidden layer, which capture the learned representations of the input sequence.

5. Experimentation and Results

The methods explained in section III and IV are applied on dataset. Below results of ML and DL algorithms are showing the performance of these algorithms.

Performance measures opted in this work for evaluating model efficiency are accuracy, precision, recall, and F1-score. These metrics assess the ability of model to correctly classify disaster and non-disaster tweets.

Figure 3 to 5 depict the performance metric of NB, SVM and RF classifiers. Each figure shows all metrics and ROC curve of respective classifier. Figure 6 and 7 illustrate model loss and model accuracy respectively for LSTM algorithm.

Comparison:

- Naive Bayes has the highest accuracy amongst the three algorithms, followed by SVM and Random Forest.

- Naive Bayes also has the highest F1-score for both classes, indicating a good equilibrium between precision and recall.

- Random Forest has the utmost recall for class 0, while Naive Bayes has the highest recall for class 1.

- SVM has comparable performance to Naive Bayes but slightly lower accuracy and F1-scores.

Naive Bayes Accuracy: 0.79
Naive Bayes Classification Report:

	precision	recall	f1-score	support
0	0.79	0.85	0.82	874
1	0.78	0.70	0.74	649
accuracy			0.79	1523
macro avg	0.79	0.78	0.78	1523
weighted avg	0.79	0.79	0.79	1523

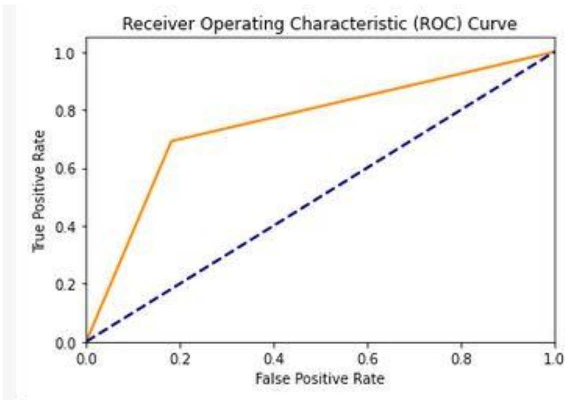

Fig.-3: Naive Bayes Classifier

Random Forest Accuracy: 0.76
Random Forest Classification Report:

	precision	recall	f1-score	support
0	0.78	0.83	0.80	874
1	0.74	0.68	0.71	649
accuracy			0.76	1523
macro avg	0.76	0.75	0.76	1523
weighted avg	0.76	0.76	0.76	1523

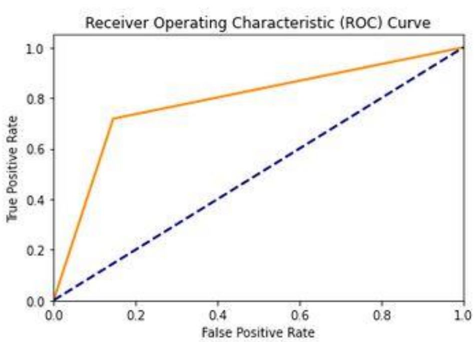

Fig.-4: Random Forest Classifier

```
SVM Accuracy: 0.77
SVM Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.82      0.80       874
           1       0.74      0.71      0.72       649

    accuracy                           0.77      1523
   macro avg       0.77      0.76      0.76      1523
weighted avg       0.77      0.77      0.77      1523
```

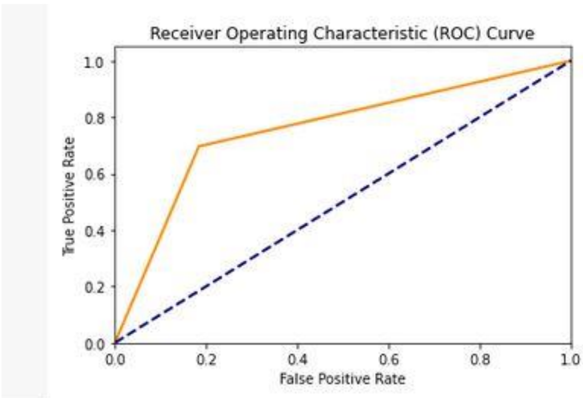

Fig.-5: Support Vector Classifier

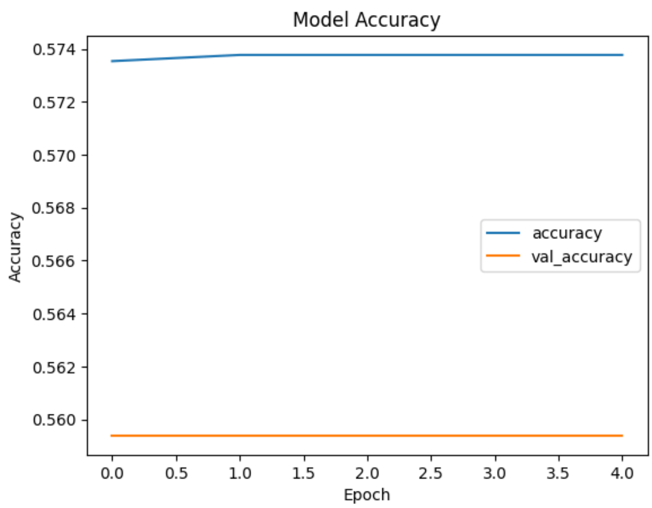

Fig.-6: LSTM Model Accuracy

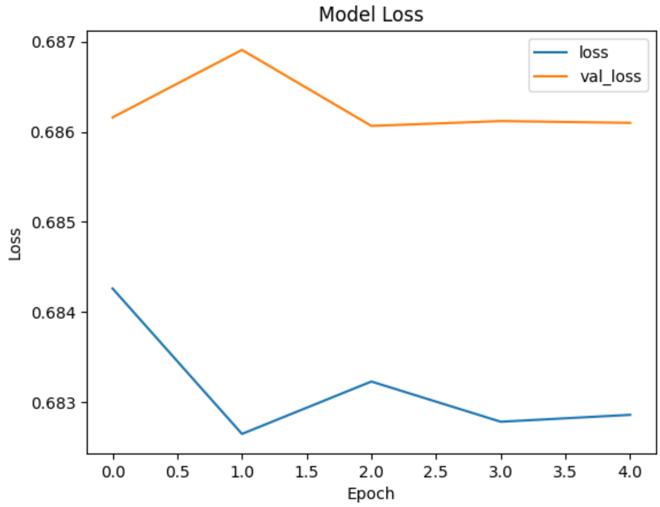

Fig.-7: LSTM Model Loss

Comments-

Based on the results of the different algorithms for sentiment analysis of Twitter messages related to disaster analysis, here's a conclusion:

1. Naive Bayes:

○ Accuracy: 80%

○ Precision, recall, and F1-score for both classes (0 and 1) are relatively balanced.

○ It shows a good overall performance with slightly better results for class 0 (negative sentiment).

2. Random Forest:

○ Accuracy: 76%

○ Precision, recall, and F1-score are slightly lower compared to Naive Bayes.

○ Like Naive Bayes, it also performs slightly better for class 0.

3. SVM:

○ Accuracy: 76%

○ Precision, recall, and F1-score are similar to Random Forest.

○ It also shows a slightly better performance for class 0.

4. LSTM:

○ Test Accuracy: 57.39%

○ The LSTM model performs the worst among the algorithms mentioned.

The accuracy is significantly lower compared to the other models, indicating that it struggles with the sentiment analysis task for disaster-related tweets.

Discussion-

Among the traditional machine learning algorithms, Naive Bayes performs the best, followed closely by Random Forest and SVM. However, for more complex tasks like sentiment analysis of tweets, deep learning models like LSTM may not always perform better compared to simpler algorithms. It's important to pick the specific attributes of the dataset and the task at hand to choose the appropriate algorithm.

6. Conclusion

This work demonstrates the potential of ML DL algorithms like Naïve Bayes, SVM, Random Forest and LSTM for disaster analysis on Twitter data. The ability to automatically classify disaster tweets can be a valuable tool for emergency response teams and humanitarian organizations. Naïve Bayes found superior in terms of accuracy and supplementary assessment metrics than other classifiers. The proposed system predicts disaster category of any real time tweet. However, it can be explored with use of pre-trained language models like BERT for performance improvement. Other types of disasters can also be modeled and integrated monitoring system can be designed for actual use.

References

1. S. Kundu, P. K. Srijith and M. S. Desarkar, "Classification of Short-Texts Generated During Disasters: A Deep Neural Network Based Approach," 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Barcelona, Spain, 2018, pp. 790-793, doi: 10.1109/ASONAM.2018.8508695.
2. Tumasjan, T. Sprenger, P. Sandner, and I. Welpe, “Predicting elections with Twitter: What 140 characters reveal about political sentiment,” Proc. Fourth Int. AAAI Conf. Weblogs Soc. Media, pp. 178–185, 2010
3. B. Joyce and J. Deng, "Sentiment analysis of tweets for the 2016 US presidential election," 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), Cambridge, MA, 2017, pp. 1-4, doi: 10.1109/URTC.2017.8284176
4. Mokhtar Ashour; CherifSalama, M. W. El-Kharashi; M. W. El-Kharashi, ”Detecting Spam Tweets using Character N-gram Features”, ICCES'18: The thirteenth International Conference on Computer Engineering and Systems, At Cairo, Egypt, 2018.
5. H. T. Phan, N. T. Nguyen, T. Van Cuong and D. Hwang, "A Method for Detecting and Analyzing the Sentiment of Tweets Containing Fuzzy Sentiment Phrases," 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Sofia, Bulgaria, 2019, pp. 1-6, doi: 10.1109/INISTA.2019.8778360.
6. Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016, “Twitter as a Lifeline: Human-annotated Twitter Corpora for NLP of Crisis-related Messages”, In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 1638–1643, Portorož, Slovenia. European Language Resources Association (ELRA).
7. Van Quan Nguyen; Hyung-Jeong Yang; Kyungbaek Kim, et.al.”Real-Time Earthquake Detection Using Convolutional Neural Network and Social Data”, IEEE Third International

Conference on Multimedia Big Data, 2017.
8. MdParvezMollah; “An LSTM model for Twitter Sentiment Analysis”. arXiv: 2212.01791v1 [cs.CL] 4 Dec 2022.
9. Pradip Bhere, Anand Upadhyay, Ketan Chaudhari, and Tushar Ghorpade, “Classifying Informatory Tweets during Disaster Using Deep Learning ”, ITM Web of Conferences 32, 03025 (2020) https://doi.org/10.1051/itmconf/20203203025, ICACC-2020
10. Zhang, Lei & Wang, Shuai & Liu, Bing. (2018). Deep Learning for Sentiment Analysis : A Survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 8. 10.1002/widm.1253.
11. G, Vinodhini & Chandrasekaran, Dr. (2012). Sentiment Analysis and Opinion Mining: A Survey. Int J Adv Res Comput Sci Technol. 2.
12. Devlin, Jacob & Chang, Ming-Wei & Lee, Kenton & Toutanova, Kristina. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
13. Yadav, Jyoti. (2023). Sentiment Analysis on Social Media. Qeios. 10.32388/YF9X04.
14. Zhu, Xiliang & Gardiner, Shayna & Roldán, Tere & Rossouw, David. (2024). The Model Arena for Cross-lingual Sentiment Analysis: A Comparative Study in the Era of Large Language Models.
15. Zhang, Kang & Geng, Yushui & Zhao, Jing & Liu, Jianxin & Li, Wenxiao. (2020). Sentiment Analysis of Social Media via Multimodal Feature Fusion. Symmetry. 12. 2010. 10.3390/sym12122010.
16. Das, Sushree & Behera, Ranjan & Kumar, Mukesh & Rath, Santanu. (2018). Real-Time Sentiment Analysis of Twitter Streaming data for Stock Prediction. Procedia Computer Science. 132. 956-964. 10.1016/j.procs.2018.05.111.
17. Lasri, Imane & Riadsolh, Anouar & Elbelkacemi, Mourad. (2023). Real-time Twitter Sentiment Analysis for Moroccan Universities using Machine Learning and Big Data Technologies. International Journal of Emerging Technologies in Learning (iJET). 18. 42-61. 10.3991/ijet.v18i05.35959.
18. V. Kharde and S. Sonawane, “Sentiment Analysis of Twitter Data: A Survey of Techniques”, International Journal of Computer Applications (0975 – 8887) Volume 139 – No.11, April 2016.
19. Sharma Sudhir & Daga, Mohit & Gemini, Bhawna. (2019), “Twitter Sentiment Analysis for Brand Reputation of Smart Phone Companies in India”. doi- 10.1007/978-3-030-30577-2_75.
20. Twitter dataset available at: https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis