Relevance Classification of Flood-Related Tweets Using XLNET Deep Learning Model

Aneela Habib, Yasir Saleem Afridi, Madiha Sher, Tiham Khan
Department of Computer Systems Engineering University of Engineering and Technology Peshawar
*Correspondence: aneelahabib2@gmail.com, yasirsaleem@uetpeshawar.edu.pk, madiha@uetpeshawar.edu.pk, tihamkhan3b@gmail.com.

Citation | Habib. A, Afridi. Y. S, Sher. M, Khan. T, "Relevance Classification of Flood-Related Tweets Using XLNET Deep Learning Model", IJIST, Special Issue pp 158-164, May 2024
DOI | https://doi.org/10.33411/IJIST/804
Received | May 10, 2024, **Revised |** May 15, 2024, **Accepted |** May 19, 2024, **Published |** May 23, 2024.

Floods, being among nature's most significant and recurring phenomena, profoundly impact the lives and properties of tens of millions of people worldwide. As a result of such events, social media structures like Twitter often emerge as the most essential channels for real-time information sharing. However, the total volume of tweets makes it hard to manually distinguish between those relating to floods and those that are not. This poses a large obstacle for responsible government officials who need to make timely and well-knowledgeable decisions. This study attempts to overcome this challenge by utilizing advanced techniques in natural language processing to effectively sort through the extensive volume of tweets. The outcome we obtained from this process is promising, as the XLNET model achieved an extraordinary F1 rating of 0.96. This high degree of overall performance illustrates the model's usefulness in classifying flood-related tweets. By leveraging the abilities of the XLNET model, we aim to provide a valuable guide for responsible governance, aiding in making timely and well-informed choices during flood situations. This, in turn, will assist reduce the impact of floods on the lives and property-affected communities around the world.

Keywords: Text classification, LSTM, Multi-head Attention, Flood, Tweets

Introduction:

Natural disasters, like floods, can cause excessive destruction to communities and residences. In the contemporary generation, social media platforms have emerged as treasured resources of statistics during and after such disasters. Twitter, in particular, plays an important position in disseminating real-time updates. However, the full extent of tweets generated through these events can overwhelm responsible governance, making it hard to identify pertinent information [1]

To deal with this problem, we advocate a text class framework making use of a machine learning model known as XLNET. XLNET is one of the latest deep learning models well-known for its splendid performance across various natural language processing (NLP) responsibilities. By leveraging XLNET's competencies, our goal is to construct an effective solution for classifying flood-related tweets. This framework could be an effective tool that can be utilized by government agencies, helping them quickly identify the relevant information amidst the vast amount of information on Twitter, thereby helping to enable appropriate and timely decision-making.

Literature Review:

Text classification is a key task in the broader framework of NLP, which aims at grouping text into predefined classes or categories. This task constitutes various jobs, such as spam detection, sentiment analysis, and theme categorization, among others. It is noteworthy, that there have been significant studies carried out on detecting natural disasters and the usage of social media and satellite imagery [2]. In recent years, social media structures, specifically Twitter, have emerged as precious assets of facts, in particular, especially during times of crisis [3]. Several studies have been carried out on taking benefit from social media for disaster reaction and management. For instance, Palen et al. (2010) [4] analyzed Twitter usage during the 2009 Red River Valley flood, identifying various sorts of tweets, together with situational recognition updates, legitimate alerts, requests for help, and emotional support. Similarly, Imran et al. (2015) [5] tested Twitter's function during the 2013 Colorado floods and highlighted its significance in supplying precious facts for disaster reaction and control.

The utility of NLP techniques in analyzing social media statistics for the duration of disasters has been widespread. Caragea et al. (2011) [6] applied machine learning methods to categorize catastrophe-related tweets, attaining an F1 score of 0.79 on a dataset comprising 9000 tweets from three exclusive disasters. Furthermore, the use of gadget-mastering models for flood detection, as demonstrated in Flood Detection in Urban Areas Using Satellite Imagery and Machine Mastering [7], signifies the ability of such procedures to improve situational attention in the course of emergencies. Extracting geographically rich know-how from micro texts like tweets becomes important for location-based structures in emergency services to correctly respond to diverse natural and man-made disasters, including earthquakes, floods, pandemics, vehicle accidents, terrorist attacks, and shooting incidents [4].

Methodology:

Figure 1 represents the block diagram of the overall methodology used in this study to illustrate the sequential steps involved in achieving our objectives. These steps are explained in the following subsections.

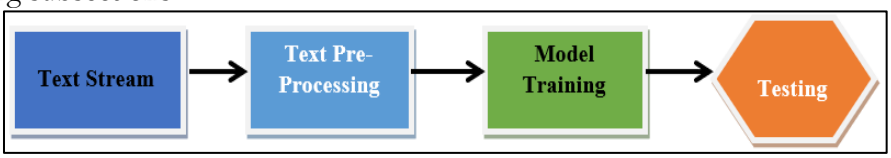

Figure 1: Block diagram of our steps to be followed

Text Stream:

In our investigation, we aim to discover the capacity of using tweets as a precious source of facts and communique during and after disasters. To accomplish this, we rent Twitter APIs

to extract relevant tweets associated with particular catastrophe events. For the purpose of model training, we've amassed and processed a substantial dataset of 5313 tweets. The preprocessing step entails cleansing the data, removing noise, and transforming the text into a suitable format that can be used for training the model efficiently.

Text Pre-Processing:

In the sphere of NLP, the pre-processing stage holds massive significance as it includes important tasks consisting of data cleansing and transformation. The objective here is to convert the raw text statistics into a format suitable for training the model [8].

Text Normalization:

Normalization of the text (the next stage in natural language analysis) normalizes the text information which later on is used to run the analysis. It is surrounded by a chain of operations that involves punctuation removal, lower-casing of all text, and special characters removal. By using these normalization techniques, we attain a greater uniform representation of the textual content, facilitating powerful evaluation and enabling NLP models, including advanced ones like XLNET, to recognize the underlying semantic content. This system lays the foundation for robust model training, complementing the overall accuracy and comprehension of the language used within the data.

Tokenization:

Tokenization can be seen as the foundational technique in NLP. It contributes a lot by working with texts such as tweets. It involves segmenting text into tokens, which are the basic units upon which most NLP processors operate, facilitating various tasks including morphological analysis. Therefore, tokenizing our tweet dataset provides the necessary structure and training for the required analysis and processing of the text. Hence, we proceed to the next step which is very essential for an actualization of the various roles within the NLP tasks and the extraction of meaningful information from the Tweets posted.

Model Training:

After preprocessing, the next step taken was the training of processed data. The choice of a 2-way type model was motivated by [8].

XLNET Model:

XLNET is a complex transformer-based language model that employs a permutation-based training approach to efficiently capture dependencies amongst all tokens within an input series. Interestingly, the architecture of XLNET is comparable with many transformer-based models like BERT. However, these models have an additional feature that sets them apart from their counterparts. Besides the above-mentioned attributes performance, this language processing model can be identified as unique and indelible in the field of NLP [9].

XLNET distinguishes itself from BERT in phrases of its training goal. While BERT relies upon left-to-right or masked left-to-proper training to predict masked tokens in a set order, XLNET employs a sequence-based training method [10]. In this method during learning, XLNET considers all viable changes of the input tokens in preference to being constrained to a specific order. As a result, XLNET captures two-way context and dependencies among all tokens in the enter dataset, main to a greater complete and nuanced learning of the language.

The architecture of XLNET is a hit fusion of the strengths discovered in automated fashions like Transformer-XL and masked language fashions like BERT. This composition boosts XLNET as the top neural language representation model, thus achieving a range of activities from NLP tasks at large [11]. This approach that shares both the input and the output contexts reinforces its ability to handle sophisticated language modalities, hence it is a proper tool in the realm of multi-lingual natural processing.

This self-attenuation structure is designed as an improvement over the conventional transformer version to overcome its limitations. The first component is the content movement representation, which is similar to the same old self-attention in Transformer. It considers both

the content x_{z_t} and the position information z_t of the target token within the input sequence as depicted in Figure 2. By doing this, the model can capture even more relevant interactions between objects and topics within the context. Another approach, termed illustration, is for this model to function as a substitute for BERT's [MASK] mechanism. It employs query circulation attention to predict the target token x_{z_t} based solely on its positional information, excluding the actual content. The model has entry to the placement information of the goal token and the context statistics before that token for making predictions.

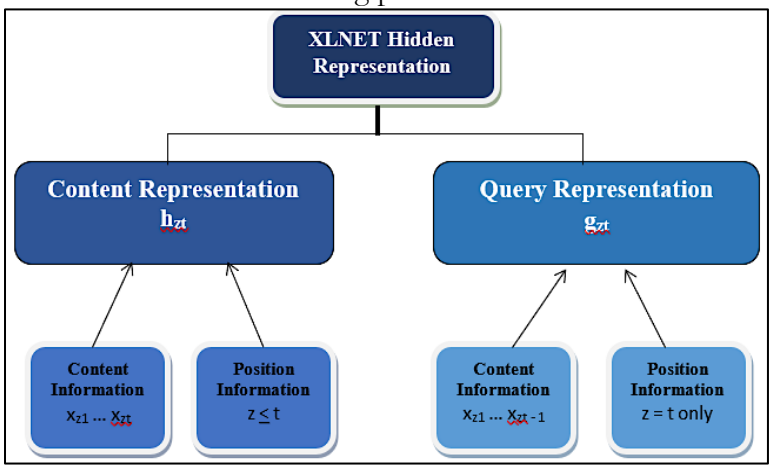

Figure 1: XLNET with 2 sets of hidden representations

By incorporating those two types of self-attention mechanisms, the Two-Stream Self-Attention architecture pursuits to better seize both local and global dependencies in the input sequence, thereby enhancing the accuracy of predictions for the target tokens. The Two-Stream attention mechanism in XLNET leads to a target-conscious prediction distribution. The key difference between XLNET and BERT lies in their pretraining strategies. Unlike BERT, XLNET does not rely on data corruption and masking during pre-training. By avoiding BERT's masking limitations, as referred to earlier in the autoencoder model, XLNET achieves its target focus and offers improved overall performance.

XLNET improves its ability to capture lengthy-variety dependencies in comparison to RNNs and general Transformers by incorporating Transformer-XL's relative encoding scheme and section repetition mechanism. The relative positional encoding is implemented primarily based on the original sequence, while the segment-level repetition mechanism prevents context fragmentation and enables the reuse of past sentence segments with new ones, thereby maintaining long-term period context. By including phase-stage repetition in hidden states, XLNET stands apart from Transformer, resulting in enhanced overall performance and better handling of remote dependencies.

XLNET integrates Transformer-XL into its improved training framework, permitting the incorporation of the repetition mechanism. This mechanism is used in the proposed permutation setting of XLNET to reuse hidden states from previous segments. However, the factorization order within the permutation from preceding segments is not cached and reused in future computations. Only the content representation of the section is retained in the hidden states, allowing for efficient handling of long-range dependencies without the need to store and reuse the whole factorization order.

LSM and Multi-Head Attention Layer: This layer is comprised of two sub-layers working together.

LSM (Likely Long Short-Term Memory): This is a type of recurrent neural network (RNN) adept at handling sequential data like text. LSTMs can capture long-term dependencies within sentences, crucial for tasks like sentiment analysis or machine translation.

Multi-head Attention:

This is a mechanism that allows the model to focus on specific parts of the input text that are most relevant to the current processing step. It essentially helps the model pay attention to different aspects of the input simultaneously.

Output Layer:

This layer takes the processed data from the previous layer and generates the final output, which could be a variety of things depending on the specific NLP task. Examples include classification which is classifying the sentiment of a text review (positive, negative, neutral), machine translation which is converting text from one language to another and text summarization which is creating a concise summary of a lengthy piece of text [12].

Overall, Figure 3 represents a simplified illustration of how a deep learning model can be structured to process and analyze textual data. By leveraging LSTMs for capturing long-term dependencies and multi-head attention to focus on relevant parts of the input, the model can learn complex patterns within the language and perform various NLP tasks.

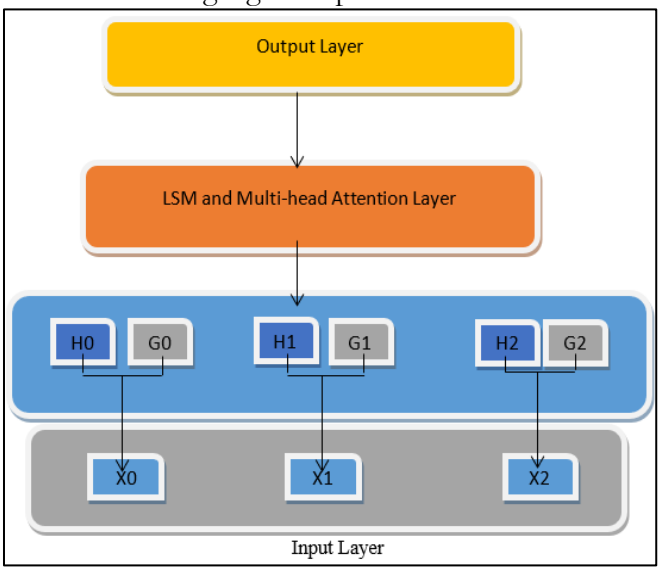

Figure 3: Layers of XLNET

Model Structure:

The XLNET model architecture can be divided into four major parts. The initial stage involves processing the entered text, where it is tokenized, meaning it is divided into individual units or words for further analysis. This process essentially operates with word vectors, resulting in low-dimensional representations of the textual input being used. Subsequently, the XLNET layer dissects the text representation and further extracts features. Such operations capture crucial patterns from the input text and then the result is transferred into Long Short-Term Memory (LSTM) and Multi-head Attention layers.

The LSTM is a type of recurrent neural network designed to capture context information from input sequences. It achieves this by using input gates, forget gates, and output gates to control the flow of information. The LSTM selects relevant memory feature vectors and integrates them to generate meaningful output. Finally, the Multi-head Attention Layer calculates the probability of attention from multiple perspectives. This lets the model assign distinctive weights to the extracted feature vectors, essentially giving greater importance to certain parts of the input text. By doing so, the model aims to acquire effective text classification.

Data Splitting:

We split our dataset into two components, training, and validation sets, as shown in Figure 4.

Training Dataset: This is a subset of the data used to train the model. The model learns

patterns and relationships from this data.

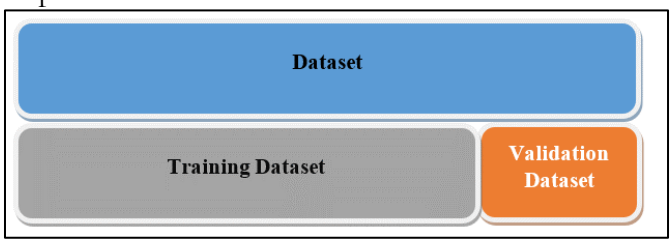

Figure 4: Splitting of the datasets

Validation Dataset:

This is another subset of the data used to assess the model's performance on unseen data. It's essential to prevent overfitting, which occurs when a model performs well on the training data but poorly on new data. The training dataset constituted 80% of the complete dataset, at the same time as the validation dataset consisted of 20% of the dataset. These separate datasets were used to train the model, investigate its performance at some point of validation, and eventually, compare its generalization on the test set.

Testing:

To evaluate the effectiveness of our model in classifying tweets as they should be, we tested it on a set of tweets. The model achieved an outstanding F1 score of 0.96 on the test dataset. This high F1 score shows that the model demonstrates sturdy overall performance and reliability in classifying tweets with an excessive stage of accuracy.

Results and Discussion:

From the results presented in Table 1, it is evident that our model achieved an accuracy of 94.16% on the test set. This accuracy rate reflects the model's ability to correctly classify tweets with a high level of precision. Looking ahead, we intend to further improve the model's performance by fine-tuning various parameters. Despite the impressive accuracy achieved, we also evaluated the model's performance using the F1 score metric, which demonstrated an outstanding score of 96%. This high F1 score indicates a robust performance in both precision and recall, underscoring the effectiveness of our model in classifying tweets accurately.

Table 1: Results of XLNET Model on test set

Method	F1 Score	Accuracy
XLNET	0.960	0.9416

Conclusion:

In our paper, we tackled the significant challenge of identifying relevant tweets after a flood occurrence, recognizing the important role of timely and accurate information for informed decision-making by governmental authorities. Leveraging advanced techniques in NLP, particularly XLNET, enabled us to efficiently sift through the immense volume of tweets generated during such events. Our findings underscore the potential of deep learning models, such as XLNET, in addressing complex challenges at the intersection of natural disasters and social media. By effectively harnessing these technologies, we have demonstrated the capability to extract valuable insights from large-scale social media data, thereby facilitating more effective disaster response strategies.

Furthermore, our research highlights the importance of ongoing innovation in the field of NLP, particularly in the context of disaster management and response. As the volume and complexity of social media data continue to grow, leveraging cutting-edge techniques like XLNET will be essential for improving the efficiency and accuracy of information extraction and decision-making processes in disaster scenarios.

References:

[1] N. Pourebrahim, S. Sultana, J. Edwards, A. Gochanour, and S. Mohanty, "Understanding communication dynamics on Twitter during natural disasters: A case

study of Hurricane Sandy," Int. J. Disaster Risk Reduct., vol. 37, p. 101176, Jul. 2019, doi: 10.1016/J.IJDRR.2019.101176.

[2] N. Said et al., "Natural disasters detection in social media and satellite imagery: a survey," Multimed. Tools Appl., vol. 78, no. 22, pp. 31267–31302, Nov. 2019, doi: 10.1007/S11042-019-07942-1/METRICS.

[3] "Using Twitter as a data source." Accessed: May 11, 2024. [Online]. Available: https://eprints.whiterose.ac.uk/126729/8/Normal_-_Ethics_Book_Chapter_WA_PB_GD_Peer_Review_comments_implemented__1_.pdf

[4] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: What twitter may contribute to situational awareness," Conf. Hum. Factors Comput. Syst. - Proc., vol. 2, pp. 1079–1088, 2010, doi: 10.1145/1753326.1753486.

[5] M. Imran and C. Castillo, "Towards a data-driven approach to identify crisis-related topics in social media streams," WWW 2015 Companion - Proc. 24th Int. Conf. World Wide Web, pp. 1205–1210, May 2015, doi: 10.1145/2740908.2741729.

[6] S. Zhang, D. Caragea, and X. Ou, "An Empirical Study on Using the National Vulnerability Database to Predict Software Vulnerabilities," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6860 LNCS, no. PART 1, pp. 217–231, 2011, doi: 10.1007/978-3-642-23088-2_15.

[7] A. H. Tanim, C. B. McRae, H. Tavakol-davani, and E. Goharian, "Flood Detection in Urban Areas Using Satellite Imagery and Machine Learning," Water 2022, Vol. 14, Page 1140, vol. 14, no. 7, p. 1140, Apr. 2022, doi: 10.3390/W14071140.

[8] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized Autoregressive Pretraining for Language Understanding," Adv. Neural Inf. Process. Syst., vol. 32, 2019, Accessed: May 11, 2024. [Online]. Available: https://github.com/zihangdai/xlnet

[9] W. J. Luo Pengcheng, Wang Yibo, "Research on automatic classification of literature subjects based on deep pre-trained language model [J]," J. Inf. Sci., vol. 10, 2010.

[10] Z. Chong, "Research on Text Classification Technology Based on Attention-Based LSTM Model [D]," Nanjing Nanjing Univ., 2016.

[11] et al. Liang Xiaobo, Ren Feiliang, Liu Yongkang, "Machine reading comprehension model based on double- layer Self-attention [J]," Chinese J. Inf., vol. 32, no. 10, pp. 130–137, 2018.

[12] S. Yu, D. Liu, W. Zhu, Y. Zhang, and S. Zhao, "Attention-based LSTM, GRU and CNN for short text classification," J. Intell. Fuzzy Syst., vol. 39, no. 1, pp. 333–340, Jan. 2020, doi: 10.3233/JIFS-191171.