Sensitive Speech analysis Using the XLNet Model

Mayank Verma
VIT Vellore
mohitsoni923@gmail.com

*Dr. Priya G
VIT Vellore
gpriya@vit.ac.in

***Abstract*— Sentiment analysis is the interpretation and classification of emotions (positive, negative and neutral) within text data using text analysis techniques. It is one of the most basic techniques in Natural Language Processing to analyze the emotions and sentiments behind a specific piece of text. In recent years, the act of terrorism has increased across many social media platforms. These platforms include Twitter, Instagram, Facebook, telegram etc. To tackle these outraging accounts and counter these users, the model presented in this paper was developed. In this paper, I will be using the concept of sentimental analysis to review tweets by numerous users provided on twitter and accurately predict which user is making a terrorism related tweet and which not.**

Introduction

In the current technological era, we have numerous tools on our disposal to make society a better place. [1]A study of twitter tweets regarding COVID Vaccine helped highlights Twitter's healthcare news, views, and criticism. Negative feelings dominated the epidemic, emphasizing the significance of mental health public health communication. The author used sentimental analysis, a method to stand as a crucial tool in decoding user sentiments and public opinions across digital platforms. The ULMFiT technique, which is available in the fastai Python module, employed the AWD-LSTM model to train and fine-tune our classifier that achieved 76.30 percent accuracy. [2]Scholars and professionals leverage cutting-edge technologies, including natural language processing and machine learning, to extract insights from textual data. An overview of a sentiment analysis pipeline is provided, encompassing data cleaning, model training, and application to diverse datasets.

Some papers used ULMFit model for sentiment analysis on twitter data. One paper did sentimental analysis of [3]US airlines tweets, the paper introduces a new method that combines ULMFiT with SVM [4]to improve sentiment detection accuracy. The model performs exceptionally well on various datasets, including achieving 99.78% accuracy on the Twitter US Airlines dataset.The ULMFit-SVM model demonstrates significant performance improvement and potential for enhancing sentiment analysis. In another paper, [5]the study investigates how well pre-trained language models perform in classifying Dutch book reviews. The research explores the impact of different training set sizes (ranging from 100 to 1600 items) on sentiment classification using ULMFiT and Support Vector Machines (SVM). ULMFiT is a superior model compared to SVM, showing better performance across different training set sizes. It achieves around 90% accuracy.

So, one thing is clear, to use ULMFit for tokenization and sentimental analysis and SVM model for training the data for high accuracy.

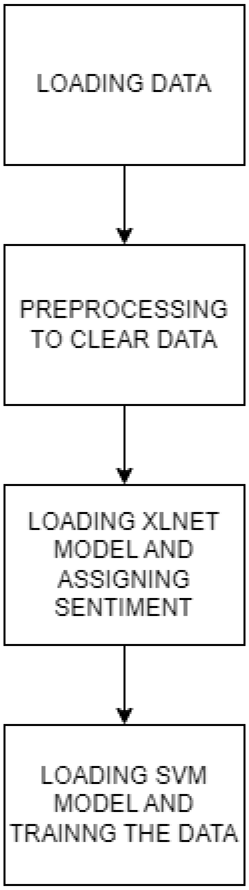
However, on doing more research, we found out that there are better models than ULMFit. [6]With the capability of modeling bidirectional contexts, denoising autoencoding based pretraining like BERT achieves better performance than pretraining approaches based on autoregressive language modeling. However, relying on corrupting the input with masks, BERT neglects dependency between the masked positions and suffers from a pretrain-finetune discrepancy. Considering these pros and cons, we propose XLNet, a generalized autoregressive pretraining method that enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and overcomes the limitations of BERT thanks to its autoregressive formulation. Furthermore, XLNet integrates ideas from Transformer-XL, the state-of-the-art autoregressive model, into pretraining. Empirically, under comparable experiment setting, XLNet outperforms BERT on 20 tasks, often by a large margin, including question answering, natural language inference, sentiment analysis, and document ranking.

So, another paper discusses COVID 19 vaccine sentiment using XLNet model. [7]The XLNet model was able to correctly classify 85% of the positive tweets and 92% of the negative tweets.

The purpose of this paper is to perform a sentiment analysis of tweets available related to terrorism and to get the best result, we will use XLNet as it outperforms ULMFit or any other available transfer learning model. This was proved by research done on multiple models.[8] Target classification schemes are built by mapping related open data in a semi-controlled manner. Target classes are built from the bottom up by DBpedia. For the experiments are used modifications of methods BERT, XLNet, Glove and ULMfit with pre-trained models for English. Two simple models with perceptron architecture are used as the baseline. The results show that the best performance for multi-label classification of DBpedia companies abstracts is achieved by XLnet models, even for unbalanced classes.

Hence, using XLNet for tokenization and SVM for training to achieve results.

Architecture:


LOADING DATA
PREPROCESSING TO CLEAR DATA
LOADING XLNET MODEL AND ASSIGNING SENTIMENT
LOADING SVM MODEL AND TRAINNG THE DATA

Picture 1

Dataset: https://www.kaggle.com/datasets/shaileshkankarej2001/dataset-of-terrorism-and-anti-terrorism-tweets

The above dataset has tweets in English along with label for identification. This dataset is used in our research.

Methodology:

1. Introduction:

   A thorough method for sentiment analysis of tweet data is included in the suggested methodology[9], [10]. The first step involves loading the raw data from a CSV file into a Pandas DataFrame. To do this, we split the data in two separate files, namely “Train.exe” and “Test.exe”. This was done to reduce the computation time and test data separately. The main goal of this process is to obtain an understanding of the sentiment that is expressed in the text.

2. Data Cleaning:

   To guarantee the integrity of ensuing analyses, data cleaning is the first step. 'clean_text', a custom function, is defined to preprocess the tweet text by eliminating unwanted elements like user names, URLs, emojis, special characters, and hashtags. To maintain uniformity, all text is also converted to lowercase. This preprocessing is done simultaneously for training and testing dataset. Results are saved as “TrainClear.exe” and “TestClean.exe”. Saving the files to manage and confirm processing.

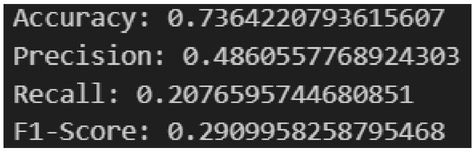
3. Sentiment Analysis using XLNet:

   The cleaned data is then loaded into a new DataFrame, for sentiment analysis. The XLNet model ('xlnet-base-cased') is employed for its proficiency in understanding contextual nuances, Tokenizing using Pre-trained models. [6]XLNet is a generalized autoregressive pretraining method that enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order. A sentiment analysis function, 'analyze_sentiment', is created to classify the sentiment of each tweet. Parallelization using the torch.multiprocessing module enhances efficiency. The sentiment labels are mapped to numerical values and appended as a new 'Sentiment' column. The resulting DataFrame is saved as “Training.exe” and “Testing.exe”

4. Sentiment Classification using SVM:

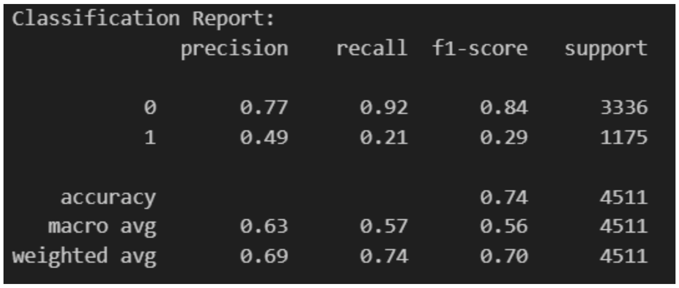
   The final step focuses on sentiment classification using a Support Vector Machine (SVM). ‘XLNet-base-cased’ model is loaded along with Training data and testing data. The two factors made are ‘X_train’, ‘Y_train’ for “Training.exe” and ‘X_test’, ‘Y_test’ for “Testing.exe”. TF-IDF vectorization is applied to convert the tweet text into numerical features, facilitating the subsequent training of an SVM model with a linear kernel. Predictions are generated on the test set, and the accuracy of the model is calculated. A comprehensive classification report is then presented, providing detailed insights into precision, recall, and F1-score metrics.

Result:



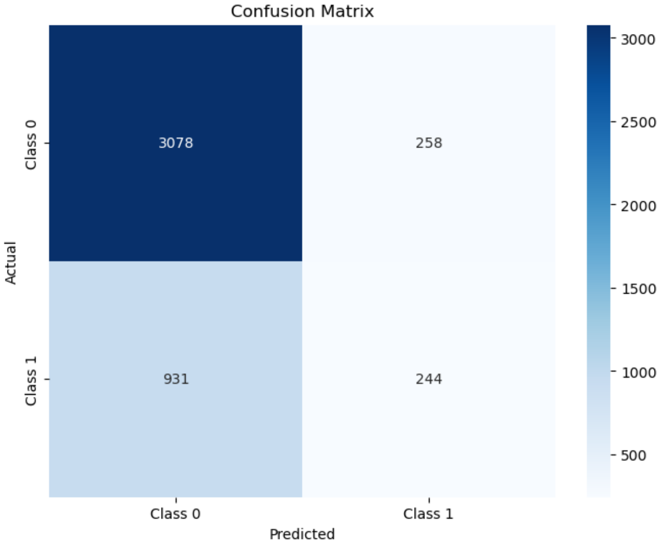
```
Accuracy: 0.7364220793615607
Precision: 0.4860557768924303
Recall: 0.2076595744680851
F1-Score: 0.2909958258795468
```

Picture 2



```
Classification Report:
              precision    recall  f1-score   support

           0       0.77      0.92      0.84      3336
           1       0.49      0.21      0.29      1175

    accuracy                           0.74      4511
   macro avg       0.63      0.57      0.56      4511
weighted avg       0.69      0.74      0.70      4511
```

Picture 3

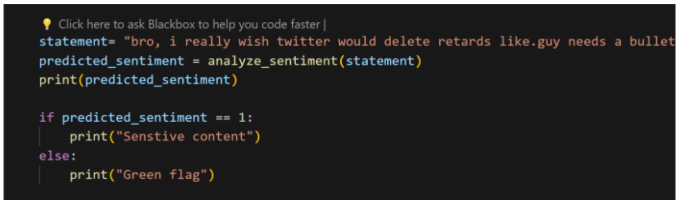


Picture 4

The classification model achieved an accuracy of approximately 73.64% on the given dataset. Precision, a measure of the model's ability to correctly identify positive instances, is around 48.61%. Recall, indicating the model's ability to capture all relevant positive instances, stands at 20.77%. The F1-score, which balances precision and recall, is at 29.10%. The classification report provides a detailed breakdown, showing that class 0 (non-terrorism) has higher precision, recall, and F1-score compared to class 1 (terrorism). The model's overall performance suggests a decent ability to distinguish between the two classes, but there is room for improvement, especially in identifying instances of terrorism.

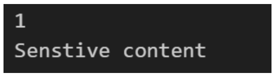
After train, we gave the model 2 strings, one positive and one negative to predict which one is Sensitive and which is not Sensitive

Input1:



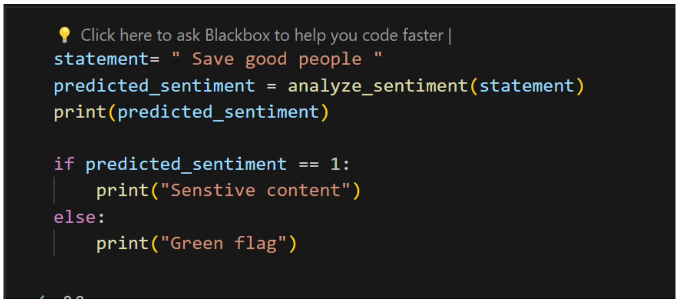
```
💡 Click here to ask Blackbox to help you code faster |
statement= "bro, i really wish twitter would delete retards like.guy needs a bullet
predicted_sentiment = analyze_sentiment(statement)
print(predicted_sentiment)

if predicted_sentiment == 1:
    print("Senstive content")
else:
    print("Green flag")
```

Output1:



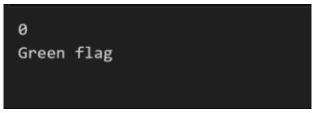
```
1
Senstive content
```

Input2:



```
💡 Click here to ask Blackbox to help you code faster |
statement= " Save good people "
predicted_sentiment = analyze_sentiment(statement)
print(predicted_sentiment)

if predicted_sentiment == 1:
    print("Senstive content")
else:
    print("Green flag")
```

Output2:



```
0
Green flag
```

Conclusion:

This study explores the important topic of sensitive speech on social media, with an emphasis on finding and analysing tweets that are connected to terrorism. In a time when the amount of this kind of content has increased on social media sites like Twitter, building strong models is essential. First, we conducted a comprehensive review of sentiment analysis methods[11], which are essential for identifying the emotional undertone present in textual data.Our approach underwent a major paradigm shift with the introduction of the XLNet model, which showed better performance than traditional techniques such as ULMFiT[12]. By utilising its bidirectional context modelling, XLNet demonstrated its adaptability to a variety of tasks, including sentiment analysis, document ranking, natural language inference, and question answering.

We employed a methodical approach that included data loading and cleaning, XLNet sentiment analysis, and Support Vector

Machine (SVM) classification. With an accuracy of roughly 73.64%, the resulting model demonstrated its effectiveness in differentiating between tweets related to terrorism and those that were not. Precision, recall, and F1-score metrics were used to identify areas for improvement in the model, even though these results demonstrate its competence in identifying instances of terrorism.

The empirical results provided information about the model's performance, highlighting its advantages and areas in need of improvement. The model demonstrated a noteworthy capacity to identify non-terrorism cases with increased accuracy, recall, and F1-score in contrast to cases involving terrorism.

Random input string tests were performed to confirm the model's applicability and confirm that it has the ability to recognise sensitive speech. As a result, this study emphasises how important sophisticated models like XLNet are to solving the problem of sensitive speech on social media. Subsequent initiatives may entail more stringent training and wider implementation on various platforms in order to augment the model's effectiveness and promote a safer virtual environment. The ongoing fight against the spread of sensitive content in the digital sphere depends on these models continuing to evolve as technology advances.

REFERENCES

[1] G. Alselwi, S. Kaynak, G. A. Abdo, and A. Alselwi, “Sentiment analysis on Covid-19 vaccines Tweets,” *researchgate.netG Alselwia, S KaynakInternational Marmara Sciences Congress (Spring), 2021•researchgate.net*, Accessed: Nov. 18, 2023. [Online]. Available: https://www.researchgate.net/profile/Ghadir-Alselwi/publication/354223609_SENTIMENT_ANALYSIS_ON_COVID-19_VACCINES_TWEETS/links/612d1cba0360302a006c6085/SENTIMENT-ANALYSIS-ON-COVID-19-VACCINES-TWEETS.pdf

[2] M. Karim and S. Das, “Sentiment Analysis on Textual Reviews,” in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Aug. 2018. doi: 10.1088/1757-899X/396/1/012020.

[3] B. AlBadani, R. Shi, and J. Dong, “A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM,” *Applied System Innovation*, vol. 5, no. 1, Feb. 2022, doi: 10.3390/asi5010013.

[4] M. Abdellatif and A. Elgammal, “Text Classification Using Language Modeling: Reproducing ULMFiT,” 2020. [Online]. Available: https://www.fast.ai/

[5] B. van der Burgh and S. Verberne, “The merits of Universal Language Model Fine-tuning for Small Datasets -- a case with Dutch book reviews,” Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.00896

[6] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V Le, “XLNet: Generalized Autoregressive Pretraining for Language Understanding”, Accessed: Nov. 17, 2023. [Online]. Available: https://github.com/zihangdai/xlnet

[7] A. Bansal, S. Susan, A. Choudhry, and A. Sharma, “Covid-19 Vaccine Sentiment Analysis During Second Wave in India by Transfer Learning Using XLNet,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13364 LNCS, pp. 443–454, 2022, doi: 10.1007/978-3-031-09282-4_37/COVER.

[8] Institute of Electrical and Electronics Engineers, *2019 Big Data, Knowledge and Control Systems Engineering (BdKCSE).*

[9] W. Medhat, A. Hassan, and H. Korashy, “Sentiment analysis algorithms and applications: A survey,” *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, Dec. 2014, doi: 10.1016/j.asej.2014.04.011.

[10] V. A. Kharde and S. S. Sonawane, “Sentiment Analysis of Twitter Data: A Survey of Techniques,” 2016. [Online]. Available: http://ai.stanford.

[11] M. Wankhade, A. C. S. Rao, and C. Kulkarni, “A survey on sentiment analysis methods, applications, and challenges,” *Artif Intell Rev*, vol. 55, no. 7, pp. 5731–5780, Oct. 2022, doi: 10.1007/s10462-022-10144-1.

[12] J. Howard and S. Ruder, “Universal Language Model Fine-tuning for Text Classification,” Jan. 2018, [Online]. Available: http://arxiv.org/abs/1801.06146