Tweet Classification Using Transformer-Based Language Models: A RoBERTa and XLNet Case Study

Flavius A. Nuta
MSc Data Science
Department of Mathematical Sciences
University of Essex
fanuta@essex.ac.uk

***Abstract*—The Natural Language Processing field has seen some significant advances in recent years, spearheaded by a number of transformer-based Language Models (LMs) such as Google's BERT and XLNet and Facebook's RoBERTa-Base. A benefit of transfer learning models is that they are pretrained on extensive datasets with concise syntax, thus allowing researchers to skip the training of the model and instead focus on fine-tuning. Despite these advancements, the question still remains on how well can pretrained models generalise noisy language full of idiosyncrasies such as social media posts. This paper attempts to address this question by proposing several data preprocessing techniques to standardise the dataset before classifying the Tweets using the RoBERTa-Base and XLNet models. The expectation is that, with the appropriate data preprocessing and fine-tuning, the models will benefit from improved accuracy.**

***Index Terms*—Natural Language Processing, Text Classification, Transfer Learning, Machine Learning, RoBERTa-Base, XLNet**

I. Introduction

Twitter is one of the most popular social media platforms, a platform which has seen an exponential rise in users over the last decade. Some of the key drivers behind this significant growth include the widespread availability of internet and the technological advancements in portable devices, for instance smartphones and tablets. undoubtedly, social media has changed the way individuals obtain information and interact with other members of society. Users are able to have discussions with people living across the globe and share content with them in real-time. While these features have shaped society, they have also opened a channel for disseminating hate speech and propagating offensive language. All of these aspects, combined with Twitter's 140 character limit that forces users to adopt concise ways of expressing ideas [11], make the platform a rich data source for analytic tasks, in particular classification.

Tweets are substantially different from more traditional forms of text such as blog posts, press articles and academic journals. They can be rife with incorrect grammar, abbreviations, hashtags that combine words together, informal vocabulary, misspellings and emojis [4], [13] - all of which can negatively impact the accuracy of Natural Language Processing models [3]. These characteristics pose certain challenges in classifying noisy Twitter data with LMs that are pretrained on well-structured text with formal vocabulary [9]. This paper will attempt to address this issue by combining a number of data preprocessing methods, alongside two separate pretrained models (RoBERTa-Base [7] and XLNet [14]), to improve their accuracy.

II. Background and Motivation

This research is motivated by the TweetEval framework [1]and its aim is to explore the possibility of improving the accuracy of RoBERTa-Base and XLNet models using more efficient data preprocessing and fine-tuning techniques. The results and performance of the models are then evaluated against the TweetEval benchmark scores[1]. To test this approach, three datasets from the TweetEval framework were selected[2]:

- **Emotion**: The dataset used to classify Tweets based on emotions was first proposed in [8] and used in [1]. Classifying text based on emotions is a task that can prove useful in a variety of areas, such as measuring customer satisfaction or targeting a marketing campaign. According to [11], connecting emotionally with a brand can motivate consumers to make a purchase. Analysing consumer's emotions with regard to a particular product can help a company understand the success behind it. Previously, this exercise was done through time-consuming methods, for example a manual review of questionnaires. However, the methods proposed in this paper can significantly reduce the timelines of this task and help companies save time and effort.
- **Hate speech**: The dataset used to identify and classify hate speech in Tweets was first proposed in [2] and used in [1]. Under the presumed anonymity of the internet, certain individuals generate hateful content which can foster discrimination against particular categories [2]. Given that the amount of content generated online increases on a daily basis, the task of detecting and fighting hate speech is becoming more relevant [2], especially since

[1]https://github.com/cardiffnlp/tweeteval#readme
[2]https://github.com/cardiffnlp/tweeteval/tree/main/datasets

regulators are putting pressure on social media platforms to remove posts that incite hatred. Making use of the methods proposed in this project can help identify hateful content online and take actions towards removing it.

- **Offensive language**: The dataset used to determine whether a Tweet contains offensive language or not was first proposed in [15] and used in [1]. Alongside the ever-growing amount of content online, offensive language is also appearing more often on social media platforms such as Twitter. Censoring offensive content with the help of human annotators can be time-consuming and it takes a toll on the mental health of those moderating such posts [15]. This reinforces the relevance of the methods proposed in this paper in detecting offensive language and combating it.

III. Methodology

The methodology used as part of this research applies a set of data preprocessing techniques to the datasets describe below, in order to prepare them for consumption. The RoBERTa-Base and XLNet models then undergo several fine-tuning experiments, and finally they are evaluated using an F1 macro-averaged score. For illustration purposes, Table 1, offers a sample Tweet for each individual dataset.

Dataset	Tweet	Label
Emotion	@user you look adorable awe	joy
Hate	#Refugees go home	hate
Offensive	@user #Rosie makes me nauseous!	offensive

Table 1: Examples of Tweets from each dataset mapped to their respective labels

A. Datasets

Emotion: This dataset contains Tweets labeled as expressing one of the following four emotions: anger, joy, sadness and optimism [1]. The models will classify the tweets from the *test* dataset into one of the four categories.

Hate speech: This dataset consists of a number of Tweets that touch upon two social issues: gender equality and immigration [1], [2]. The models will evaluate each Tweet from the *test* dataset and assign one of the following labels: hate or not-hate.

Offensive language: This dataset consists of a number of Tweets, some of which may contain offensive language [1], [15]. The models will evaluate whether an individual Tweet contains offensive language and then assign it one of the following labels: offensive or not-offensive.

Table 2, illustrated below, presents the relevant splits across all datasets.

Dataset	Labels	Train	Test
Emotion	4	3,257	1,421
Hate Speech	2	9,000	2,970
Offensive lg.	2	11,916	860

Table 2: Datasets split

B. Data Preprocessing

Data preprocessing is one of the key proposals of this research. Content shared on Twitter often contains abbreviations, misspelled words, incorrect grammar, emojis and unnecessary punctuation marks. The codes developed in this project process the raw data in order to remove as much noise as possible, with the ultimate goal of improving model accuracy, as described below:

- **Misspelled words**: Tweets can contain misspelled words such as 'devide' instead of 'divide'. To remediate these inconsistencies, the project has adopted a dictionary[3] containing 2,455 misspellings of 1,922 words. The code uses the dictionary to replace the incorrect form with the appropriate spelling.
- **Lowercase words**: All text is converted to lowercase to ensure uniformity throughout the whole process, so that the word vector is not affected by the capitalisation of a word.
- **Abbreviations**: Text on social media often contains abbreviations of words or expressions, such as 'ASAP' instead of 'as soon as possible', or 'bcuz' instead of 'because'. To cater for this issue, the project has created a dictionary containing over 70 abbreviations. The code is used to replace the identified abbreviations with the full version of the expression or word.
- **Removing HTML elements**: Data scraped from websites, for instance Twitter, can contain various HTML elements. A few examples are '&' which represents the ampersand symbol ('&'), or '<' for the 'less-than' sign. To streamline the tokenisation process, these elements are removed with the help of the Python *html* module.
- **Replacing emojis with text**: Emojis are very popular amongst social media users and therefore play a significant role in any social media ecosystem. Emojis have come to depict numerous emotions, objects or feelings and can provide valuable insights about the essence of a message. This project has adopted a dictionary[4] to translate any identified emoji into text.
- **Replacing language contractions**: Due to a combination of fast-typing and limited character count per Tweet, users rely on contractions to express their thoughts. This can affect the tokenisation of words, hence why expressions such as "you're" are replaced with the separate words on which they are based [11].
- **Removing user mentions**: Twitter users often mention one or more other users using a handle. All user mentions are removed from the text.
- **Punctuation marks**: Exclamation and question marks reinforce specific feelings, emotions or tones, thus making a statement more impactful [11]. Following the approach described in [11], these punctuation marks are replaced with their literal meaning. Any other unnecessary punc-

[3] http://www.dcs.bbk.ac.uk/ROGER/corpora.html

[4] https://github.com/NeelShah18/emot/tree/master/emot

tuation marks are removed, including hashtags, commas and periods.
- **Tokenisation**: The project uses the Python *simpletransformers* module to tokenise the text as part of the models setup[5].

C. Classification Models

The Transformer-based LMs XLNet [14] and RoBERTa-Base [7] have changed the Natural Language Processing landscape by outperforming neural network models, such as CNN and LSTMs, in classifying text [12]. This comes as a result of the XLNet and RoBERTa-Base models being pretrained on large text corpora.

RoBERTa-Base is a bi-directional transformer pretrained over a significant quantity of unlabeled textual data to learn a language representation that can be used to fine-tune for specific machine learning tasks. Pretraining for RoBERTa relied on 160 GB of data, split as follows:

- **BooksCorpus** [16] and English Wikipedia data, adding up to 16 GB;
- **CommonCrawl News** dataset, containing 63 million English language news articles. These add up to 76 GB [7];
- **OpenWebText** dataset, containing 36 GB of Reddit data [7]; and
- **Stories** dataset, containing a subset of CommonCrawl adhering to the Winograd schemas [10], which amounts to 31 GB of data.

XLNet is a large, bidirectional transformer that uses an improved training methodology, larger datasets and more computational power to improve prediction accuracy. XLNet revolutionised the field of Natural Language Processing by introducing permutation language modeling, where all tokens are predicted but in random order [14]. XLNet contrasts traditional language models where all tokens are predicted in sequential order, instead of random order. This approach allows the model to learn bidirectional relationships and to better handle dependencies and relations between words. Pretraining for XLNet relies on 129 GB of data, split as follows:

- **BooksCorpus** [16] and English Wikipedia data, adding up to 16 GB;
- **CommonCrawl**, **Giga5** and **ClueWeb** 2012 data, adding up to 113. GB [14]

D. Model Fine-tuning

Each model is fine-tuned in line with the recommendations made by their authors. Despite the models being pretrained, the paper also looks at conducting additional training on the datasets shown in Table 2. For RoBERTa, several runs have been performed with various combinations of the hyperparameters that are outlined below [7]:

- **Training batch size**: 8, 16, 32, 64, 128
- **Evaluation batch size**: 8, 16, 32, 64, 128
- **Learning rates**: 5e-5, 4e-5, 3e-5, 2e-5, 1e-5
- **Epochs**: 2, 3, 4
- **Adam optimiser epsilon value** [5]: 1e-8, 2e-8, 3e-8
- **Maximum sequence length**: 128 tokens

For XLNet, several runs have been performed with various combinations of the hyperparameters that are outlined below:

- **Training batch size**: 8, 16, 32, 64
- **Evaluation batch size**: 8, 16, 32, 64
- **Learning rates**: 5e-5, 4e-5, 3e-5, 2e-5, 1e-5
- **Epochs**: 2, 3, 4
- **Adam optimiser epsilon value** [5]: 1e-8, 2e-8, 3e-8
- **Maximum sequence length**: 128 tokens

E. Evaluation Metrics

To evaluate the classification performance, the project will use a macro-averaged **F1 Score**, as described in the TweetEval framework [1]. The formula for the F1 Score can be defined as the harmonic mean of the two measures [6]:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} (1)$$

The F1 Score is the preferred measure to rate the accuracy of a classifier, as we are looking for a balanced measure between precision and recall.

IV. Results

A. Overview

The F1 scores for the two models across the three tasks are presented in Table 3. At a high level, it is observed that the RoBERTa-Base model has outperformed XLNet by approximately 4% on both the *Emotion* and *Hate* classification tasks, while the XLNet has been more accurate at detecting *Offensive* language.

Model	F1 Emotion	F1 Hate	F1 Offensive
RoBERTa-Base	**83.16%**	**61.24%**	80.38%
XLNet	79.48%	56.90%	**81.40%**

Table 3: High Level Results

B. Model Specifications

1) Emotion Task: For RoBERTa-Base, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **16**, an *evaluation batch size* of **16**, a *learning rate* of **3e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **2e-8**. The following confusion matrix was obtained after running the RoBERTa-Base model.

[5]https://github.com/ThilinaRajapakse/simpletransformers

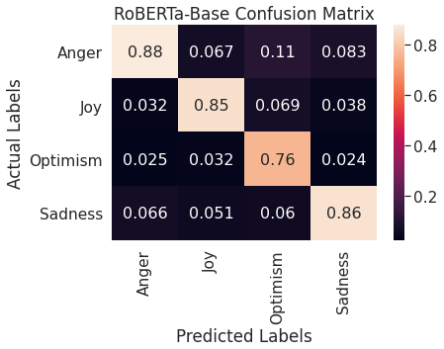

Figure 1: Normalised RoBERTa-Base Emotion Confusion Matrix

The confusion matrix (Figure 1) shows RoBERTa-Base performed well in predicting *anger*, *joy* and *sadness* emotions, with scores above the overall F1 metric. The model did underperform in identifying *optimism*, with 76% of labels being assigned correctly.

For XLNet, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **16**, an *evaluation batch size* of **16**, a *learning rate* of **3e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **1e-8**. The following confusion matrix was obtained after running the XLNet model.

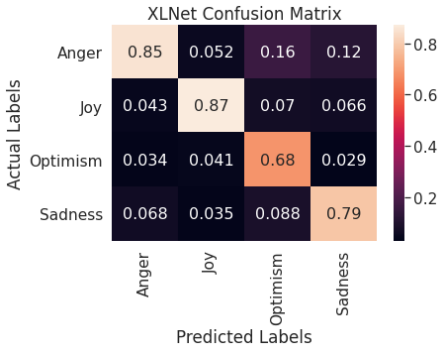

Figure 2: Normalised XLNet Emotion Confusion Matrix

The confusion matrix (Figure 2) shows XLNet performed well in predicting *anger* and *joy*, with results being similar to those obtained by the RoBERTa-Base model. XLNet did, however, fall behind in identifying *optimism* and *sadness*.

2) Hate Task: For RoBERTa-Base, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **128**, an *evaluation batch size* of **128**, a *learning rate* of **2e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **2e-8**. The following confusion matrix was obtained after running the RoBERTa-Base model.

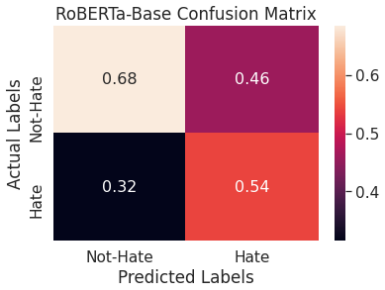

Figure 3: Normalised RoBERTa-Base Hate Confusion Matrix

The confusion matrix (Figure 3) shows RoBERTa-Base performed better when predicting *not-hate*, with a score above the overall F1 metric. The model did underperform in identifying *hate*.

For XLNet, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **64**, an *evaluation batch size* of **64**, a *learning rate* of **2e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **2e-8**. The following confusion matrix was obtained after running the XLNet model.

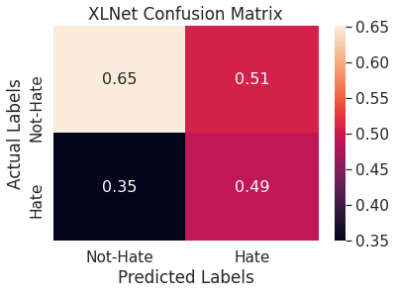

Figure 4: Normalised XLNet Hate Confusion Matrix

The confusion matrix (Figure 4) shows XLNet performed almost as well as RoBERTa-Base in predicting *not-hate*. Similarly, XLNet fell behind in identifying *hate* statements.

3) Offensive Task: For RoBERTa-Base, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **128**, an *evaluation batch size* of **8**, a *learning rate* of **1e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **2e-8**. The following confusion matrix was obtained after running the RoBERTa-Base model.

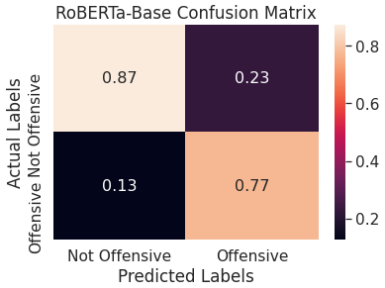

Figure 5: Normalised RoBERTa-Base Offensive Confusion Matrix

The confusion matrix (Figure 5) shows RoBERTa-Base performed well when predicting *not-offensive* language, with 87% of labels being assigned correctly. The model did underperform in identifying *offensive* language, with only 77% of labels being assigned properly.

For XLNet, the optimal settings involved fine-tuning the model hyperparameters to cater for a *training batch size* of **64**, an *evaluation batch size* of **16**, a *learning rate* of **1e-5**, **4** *epochs* and an *Adam optimiser epsilon* value of **2e-8**. The following confusion matrix was obtained after running the XLNet model.

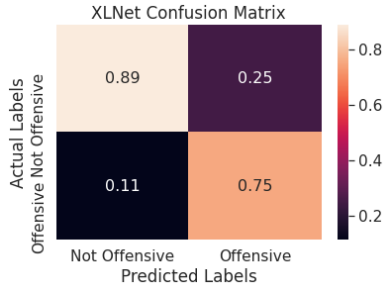

Figure 6: Normalised XLNet Offensive Confusion Matrix

The confusion matrix (Figure 6) shows XLNet performed better than RoBERTa-Base when identifying *not-offensive* language. XLNet did, however, fall slightly behind in identifying *offensive* language, with 75% of labels being assigned correctly.

V. Discussion and Results Comparison

The results obtained as part of this research are evaluated against the TweetEval framework to determine whether more complex data preprocessing and model fine-tuning have an impact on prediction accuracy. The RoBERTa-Base and XLNet models have outperformed the scores obtained in the TweetEval framework across all three datasets.

A. Emotion Task Results Comparison

Model	TweetEval Results	Current Results
RoBERTa-Base	76.10%	**83.16%**
XLNet	-	79.48%
BERTweet	79.30%	-
RoBERTa-Retrain	78.50%	-
RoBERTa-Twitter	72.00%	-
FastText	65.20%	-
LSTM	66.00%	-
SVM	64.70%	-

Table 4: Results Comparison - Emotion

The RoBERTa-Base model employed as part of this research has outperformed the highest ranked TweetEval model by 4% and its TweetEval counterpart by 7%. XLNet has also outperformed all of the TweetEval models for the Emotion task.

B. Hate Task Results Comparison

Model	TweetEval Results	Current Results
RoBERTa-Base	46.60%	**61.24%**
XLNet	-	56.90%
BERTweet	56.40%	-
RoBERTa-Retrain	52.30%	-
RoBERTa-Twitter	49.90%	-
FastText	50.60%	-
LSTM	52.60%	-
SVM	36.70%	-

Table 5: Results Comparison - Hate

The RoBERTa-Base model employed as part of this research has outperformed the highest ranked TweetEval model by 5% and its TweetEval counterpart by 15%. XLNet has also outperformed all of the TweetEval models for the Hate task.

C. Offensive Task Results Comparison

Model	TweetEval Results	Current Results
RoBERTa-Base	79.50%	80.38%
XLNet	-	**81.40%**
BERTweet	79.50%	-
RoBERTa-Retrain	80.50%	-
RoBERTa-Twitter	77.10%	-
FastText	73.40%	-
LSTM	71.70%	-
SVM	52.30%	-

Table 6: Results Comparison - Offensive

The RoBERTa-Base model employed as part of this research has outperformed all of the TweetEval models with the exception of RoBERTa-Retrain. XLNet on the other hand has outperformed all TweetEval models.

VI. Conclusion

Based on the results obtained, it has been identified that more complex data preprocessing techniques and model fine-tuning achieve substantial improvements to model accuracy, leading to an improvement in predictions of up to 15%. The research achieves state-of-the-art results, illustrating the importance of these previously overlooked design decisions. Additional improvements can be made to some of the preprocessing techniques, particularly when it comes to misspelling and abbreviations by expanding the dictionaries used. Future avenues for research include expanding the designs proposed in this paper to perform multi-task learning (e.g. exploiting the similarities between hate speech identification and offensive language detection to derive new insights).

REFERENCES

[1] Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa-Anke, and Leonardo Neves. TweetEval:Unified Benchmark and Comparative Evaluation for Tweet Classification. In *Proceedings of Findings of EMNLP*, 2020.

[2] Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA, 2019. Association for Computational Linguistics.

[3] Jacob Eisenstein. What to do about bad language on the internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 359–369, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

[4] Yuheng Hu, Kartik Talamadupula, and Subbarao Kambhampati. Dude, srsly?: The surprisingly formal nature of twitter's language. In *Proceedings of the 7th International Conference on Weblogs and Social Media, ICWSM 2013*, pages 244–253. AAAI press, 2013. 7th International AAAI Conference on Weblogs and Social Media, ICWSM 2013 ; Conference date: 08-07-2013 Through 11-07-2013.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[6] Zachary Chase Lipton, Charles Elkan, and Balakrishnan Narayanaswamy. Thresholding classifiers to maximize f1 score, 2014.

[7] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[8] Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17, 2018.

[9] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. Bertweet: A pre-trained language model for english tweets, 2020.

[10] Trieu H. Trinh and Quoc V. Le. A simple method for commonsense reasoning, 2019.

[11] Parth Vora, Mansi Khara, and Kavita Kelkar. Classification of tweets based on emotions using word embedding and random forest classifiers. *International Journal of Computer Applications*, 178(3):1–7, Nov 2017.

[12] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[13] Wei Xu. From shakespeare to Twitter: What are language styles all about? In *Proceedings of the Workshop on Stylistic Variation*, pages 1–9, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[14] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2020.

[15] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, 2019.

[16] Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books, 2015.