A BERT-based Hate Speech Classifier from Transcribed Online Short-Form Videos

Rommel H. Urbano Jr.
College of Computing and Information Technologies, National University-Manila, Philippines

Jeffrey U. Ajero
College of Computing and Information Technologies, National University-Manila, Philippines

Angelic L. Angeles
College of Computing and Information Technologies, National University-Manila, Philippines

Maria Nikki H. Quintos
College of Computing and Information Technologies, National University-Manila, Philippines

Joseph Marvin R. Imperial
College of Computing and Information Technologies, National University-Manila, Philippines

Ramon L. Rodriguez
College of Computing and Information Technologies, National University-Manila, Philippines

ABSTRACT

With the rise of human-centric technologies such as social media platforms, the amount of hate also continues to grow proportionally with the increasing number of users worldwide. TikTok is one of the most-used social media platforms due to its feature that allows users to express themselves via creating and sharing short-form videos based on any desired topic and content. In addition, it has also become a platform for political discourse and mudslinging as users can freely express an opinion and indirectly debate with random people online. In this study, we propose the use of BERT, a complex bidirectional transformer-based model, for the task of automatic hate speech detection from speech transcribed from Tagalog TikTok videos. Results of our experiments show that a BERT-based hate speech classifier scores 61% F1. We also extended the task beyond several algorithms such as LSTM, Naïve Bayes, and Decision Tree and found out that traditional methods such as a simple Bernoulli Naïve Bayes approach remain at par with the BERT model.

CCS CONCEPTS

• **Computing methodologies** → Machine learning; Learning paradigms; Supervised learning; Supervised learning by classification;

KEYWORDS

Hate Speech, TikTok, Filipino Language, Bidirectional Encoder Representations from Transformers (BERT)

ACM Reference Format:
Rommel H. Urbano Jr., Jeffrey U. Ajero, Angelic L. Angeles, Maria Nikki H. Quintos, Joseph Marvin R. Imperial, and Ramon L. Rodriguez. 2021. A BERT-based Hate Speech Classifier from Transcribed Online Short-Form Videos. In *2021 5th International Conference on E-Society, E-Education and E-Technology (ICSET 2021), August 21–23, 2021, Taipei, Taiwan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3485768.3485806

1 INTRODUCTION

Since we are living in the 21st century where different technological advancements have taken place, that contributes to the growth of crimes and hate as more and more people are expressing their views and opinions through social media platforms such as Facebook, Twitter, and TikTok. Thus, like social media content and usage increase, so does the growth of hate speech. Some social media contents are enlightening and informative, others are humorous and interesting, while others are still political or religious. This is where technological advancements such as social media begin to play a catalytic role in hate speech.

TikTok is widely known as a fun and entertaining social media platform that allows users to create and share a variety of short-form videos based on their desired topic and content, along with its special effects and filters ranging from viral dances, comedy videos, and challenges. Recently, there has been an increased volume of platform content relating to hate speech in the context of politics. TikTok users or whom they label as "TikTokers" can influence and impact our society, both positively and negatively. *TikTokers* can positively impact the users by using it as a tool to educate other users on important topics, spread awareness (e.g., mental health awareness, environmental matters, charity campaigns), gain peers and moral support, explore their talents, and boost self-confidence. TikTok is an entertainment platform but also has a negative side which can negatively impact the users by TikTokers spreading hate and inappropriate content (e.g., body shaming, trolls, racism, and profanity). In addition, TikTok has become a platform for political discourse and trolling since we all have the freedom to express an opinion and debate with a random person online which often leads to hate and sarcasm.

Hate speech is said to be a direct attack on people based on what is known as the "protective characteristic" — race, ethnicity, national origin, religious affiliation, sexual orientation, sex, gender, gender identity, or serious disability or disease [1]. It is worth mentioning that although some platforms are restricting and implementing policies to eliminate hate speech content, this is still considered a difficult task. Since the spread of hate speech nowadays is through social media platforms, the importance of the study of automatic detection of hate speech is important as it reduces the spread of hateful content that may add up to hate crimes which will negatively impact our society.

ICSET 2021, August 21–23, 2021, Taipei, Taiwan
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-9015-6/21/08…$15.00
https://doi.org/10.1145/3485768.3485806

Over the years, different technologies have been used to conduct several studies on automatic hate speech detection including transformer-based models such as the most successful machine learning technique, Bidirectional Encoder Representation from Transformers (BERT) that solved various Natural Language Processing (NLP) tasks [2] [3] [4]. Furthermore, unlike common languages such as English and Spanish, low-resource languages such as Filipino suffer from a lack of expert-annotated corpus or readily available resources and a dataset which makes it a challenging task. In addition, transformer-based models on low-resourced Filipino language have been explored by few researchers and are proven to be effective on low-resource tasks [5] [6] [7].

As hate speech continues to grow, so does its negative impact that contributes more to the problem in our society. The demand for automatic hate speech detection systems becomes more evident and significant. Given this problem, hate speech detection remains an active area of research. In addition, based on the related studies we gathered, it is apparent that there is limited work for automatic hate speech detection in TikTok despite being one of the largest social media platforms. In this study, we describe our following contributions:

- We propose the use of BERT for the task of automatic hate-speech detection from 1,000 manually annotated Tagalog online short-form videos (TikTok) through manual video transcription. The video transcription is done manually to avoid inaccuracy in the dataset. The dataset collected contains two primary classes which are hate and non-hate.
- Interpretation of BERT predictions from the annotated data for hate-speech detection.
- Contribute to the exploration on detecting hate speech in low-resource languages such as Filipino using BERT, and at the same time, present a useful reference for future studies.

2 RELATED LITERATURE

Since the research on automatic hate speech detection is not new, and researchers over the years aim to have an accurate hate speech detection tool, different approaches and implementations are proposed to reduce the number of hate content through social media platforms (i.e., tweets, news outlets online, TikTok, etc.). Different established approaches made by the researchers include the use of common machine learning algorithms such as a simple unidirectional 1- layer LSTM model [8] for hate speech detection on Spanish and English language datasets. Another method is the use of a semantic feature extractor (i.e., Syntactic, Semantic, Contextual) that utilizes an SVM classifier [9] on 1000 scraped tweets. In addition, utilizing tweets that were manually classified into 3 classes (i.e., hateful, offensive, and clean) is also used on a common machine learning algorithm, such as the use of Naive Bayes, Random Forest [10], SVM, KNN, Decision Tree, Adaptive Boosting, Multilayer Perceptron, and Logistic Regression [11], an N-gram and TFIDF based Approach [12], and a Linear SVM classifier that uses lexical features [13].

In addition, research studies about hate speech detection have been robust, and more studies were conducted over the years. This leads to unlocking more possibilities in computing, which introduces us to Deep Learning approaches. Since deep learning is the successor of Machine Learning, Rule-Based Machine Learning, and a common Deep Learning approach (i.e., Feed Forward NN) [14] is utilized on Hate Speech Detection using Philippine Election Data from Twitter tweets. While Recurrent Neural Network (RNN) [15] was used on Web resource that contains a monolingual corpus of different languages created from translations of the Bible which used only the Book of Genesis in Cebuano and Tagalog versions.

Similarly, a state-of-the-art transformers-based machine learning technique, BERT and DistilBERT [6] were utilized using tweets during the 2016 Philippine Presidential Election debates and tweets related to the 2016 election hashtags. Another resembling approach was done by implementing a pre-trained BERT model [3] using a Twitter Dataset [17] which contains 16,000 tweets based on an initial ad-hoc approach that searched common insults and terms related to religion, sex, gender, and ethnic minorities. In addition, another dataset is also used which contains 84.4 million tweets from 33,458 Twitter users [18] which consist of terms from a pre-defined lexicon of hate speech phrases and words known as Hatebased.org.

On the other hand, since Filipino is a low source language, transfer learning on AWD-LSTM [19] has been proposed to test the capabilities of the model using a hate speech dataset that contains 1,000, 5,000, and 10,000 samples. To alleviate resource scarcity in the Filipinos, an automated method in producing natural language Inference benchmark datasets in the Filipino language was proposed using pre-trained Transformers based on the ELECTRA pretraining scheme [5]. Moreover, an automatic white supremacist hate speech detection using two approaches which are domain-specific word embeddings with deep learning and natural language processing techniques such as BERT on the Twitter dataset, Stormfront dataset, and combined dataset from Twitter and Stormfront was conducted [4].

Furthermore, different model fine-tuning techniques such as BERT and ULMFiT are utilized in a low-resourced Filipino language using a benchmark language modeling Filipino dataset known as WikiText-TL-39 [7]. A similar approach has been done but using BERT only, this study proves that BERT is effective for both fine-tuning and feature-based approaches on 11 NLP tasks across all 4 datasets which are The General Language Understanding Evaluation (GLUE) dataset, The Stanford Question Answering Dataset (SQuAD v1.1), The SQuAD 2.0 dataset which extends the SQuAD 1.1, and The Situations with Adversarial Generations (SWAG) dataset [2].

3 METHODOLOGY

Focusing on low resource languages such as Filipino/Tagalog pushes forward the current state of the art in low resource languages. The research in the field of the Filipino/Tagalog language needs to keep up and set baselines for further researchers to build upon and improve. Inline to detect hate speech in TikTok video transcriptions, the researchers conducted several tasks in building the dataset and developing the prediction model.

3.1 Data Collection

This study's data was collected using an unofficial API wrapper for TikTok called *TikTok-API* [20], and a free open-source download management tool called *Jdownloader2* [21]. The researchers first

Table 1: Data Annotation

Video Name	Voter 1	Voter 2	Voter 3	Final Vote
Video 1	0	0	0	0
Video 2	0	0	1	0
Video 3	1	1	1	1

selected TikTok content creators that are known for uploading hate content to load into the *TikTok-API* for the hate speech data, before selecting another batch of creators that are known for uploading educational and comedy skit styled content for the non-hate data. After selecting various content creators, *TikTok-API* then provided the download links of the selected creators' videos in a CSV format that the researchers loaded to *Jdownloader2* which downloaded the videos automatically one by one. After a total of 4,746 downloads, only 1,000 videos were kept after the manual filtering process, where the only considered videos are that of:

- Does not contain or has minimal background music.
- Only uses Filipino or *Taglish* (Tagalog-English) language.
- Does not have more than one face on the frame at the same time and should be visible.
- Has no lip-syncing.

3.2 Data Annotation and Transcription

In labeling 1000 filtered and collected videos corresponding to their classification, manual review, and annotation of each video was conducted. The review of each video is conducted by 3 annotators and each of the annotators is to watch one video. Afterward, annotators are to vote whether the video watched is *hate speech* or *non-hate speech*. In addition, after the annotation, the text is extracted from the video via manual transcription. The criteria in determining *hate speech* videos are as follows:

- TikTok videos attacking/mocking certain beliefs/ideals.
- TikTok videos offensively replying to user comments (replies of ad hominem and such).
- TikTok videos attacking a group of people.
- TikTok videos attacking/provoking a specific person.

If the video preview does not meet the criteria of *hate speech*, the video is automatically annotated as *non-hate speech*. The annotation of the 1000 videos was conducted by 12 people, with each video having 3 annotators. The voting only includes 0 being *non-hate speech* and 1 being *hate speech*. Once the votes of the 3 annotators are complete for a specific video, a *final vote* of what classification does the video belongs to is decided through a majority decision. Table 1 shows the sample annotation from the 1000 video annotation records.

A similar process was conducted on *Multi-Class Twitter Emotion Classification: A New Approach* [9] wherein the researchers also conducted manual annotation of their generated dataset and conducted a *Cohen Kappa* computation in gauging the extent of consensus between 2 annotators. This study utilizes 3 annotators for each video, thus, instead of using *Cohen Kappa* as the annotation agreement metric, with 3 annotators, *Fleiss Kappa* is used. Using NLTK's agreement library [22], the *Fleiss Kappa* computation score results in a 0.75 or 75% inter-annotation agreement.

After the annotation of the dataset, manual transcription of each video was conducted. 12 people were tasked to manually transcribe the 1000 TikTok videos collected on a CSV file, and each transcription corresponds to the classification assigned by the annotators. The generated TikTok transcription dataset contains 3 columns: ID, Video Name, text (transcription), and Final Vote (from data annotation/majority decision). Table 2 shows the sample data from the generated dataset.

3.3 Data Pre-Processing

In line with the goal of hate speech detection on TikTok Video Transcriptions, pre-processing of data is kept to a minimum to preserve the context of each transcription. The "ID" and "Video Name" columns of the dataset are dropped during preprocessing retaining only the "text" and "Final Vote" columns. The performed *data cleaning* on the manually generated dataset is *Lowercasing* and *Punctuation Removal. Lowercasing* the text transcriptions will allow words of the same instance but different in casing to represent a single canonical form (ex: 'You' and 'you'). The same goal is observed with *Punctuation Removal*, in general, encoding words with attached punctuations is represented differently on words without punctuations, for example, the word 'you' is represented differently with the word instance 'you!'. Removing punctuations on the text transcriptions also allows preserving the context of the text, at the same time keeping a single canonical form of each word instance during encoding/embedding.

In transforming our processed dataset into numerical forms, preprocessing includes the use of pre-trained BERT Tokenizers. Words on each transcription record are tokenized using a pre-trained *Filipino BERT Tokenizer* [7] to match the requirements of the model architecture. Pre-trained BERT Tokenizer includes having 100 max length tokens per instance, which outputs the tokenized words and attention masked records.

Data splitting includes several variations of training, validation, and testing partitions. Table 3 shows the different train-validation-test splits conducted. Aside from the traditional training-validation-testing partitioning, K-fold Cross Validation is also conducted in data partitioning. To add, class weights are balanced for the experiment.

3.4 Model Architecture

For the Hate Speech detection task on TikTok video text transcription, the researchers utilized *transfer learning* in the form of a pre-trained Filipino BERT [7]. The use of BERT for this task offers several advantages as per generating prediction models from scratch. First, several research works suggest that the use of BERT is advantageous in sentiment analysis tasks [7] due to the algorithm's capability of learning the language. Second, using the Filipino BERT [7] not only accommodates the commonality of using Filipino/Tagalog language for the classification task but also accommodates the low number of data available in the generated dataset. Lastly, the training time of the BERT model is faster than of implementing other deep learning classification algorithms from scratch, since the Filipino BERT is already pre-trained on Filipino Datasets

Table 2: Dataset Snippet

ID	text	Final Vote
483	Tama nga naman, biblical ang pagdi-disiplina. Kaya lang depende kung. . .	0
484	Naks, ano nga pala ang rule nyo nay sa bansa natin?...	1
485	Okay, may nag comment. Another dilawan spitted. Ano daw?...	1

Table 3: Train-Val-Test Split

Split Partition(train-val-test)	Train Data Count	Validation Data Count	Test Data Count	Total Data Count
90-5-5	900	50	50	1000
60-20-20	600	200	200	1000
70-10-20	700	100	200	1000

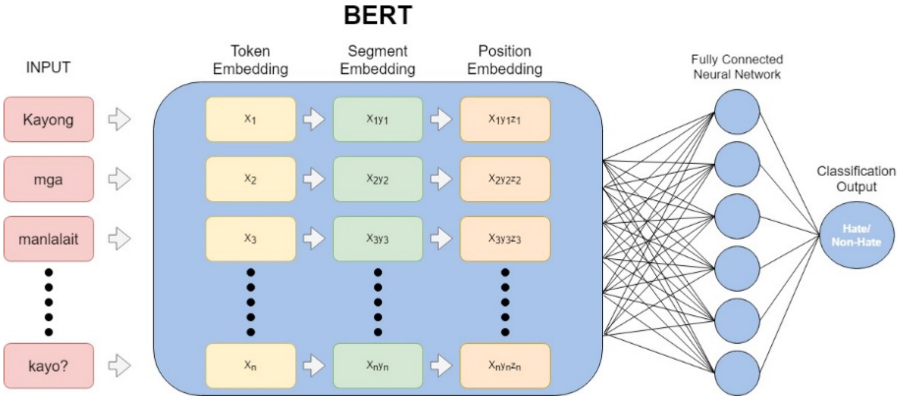


Figure 1: BERT Architecture

which leaves only fine-tuning of the model to fit the hate speech detection task on TikTok video transcriptions. For the parameters and optimizer used in the experiment: 0.1 dropout rate was used, *Rectified Linear Unit* (ReLU) activation function in between hidden layers was utilized, and a *SoftMax* activation function for the output layer. The optimizer used is *Adam Optimizer* with a 0.001 learning rate parameter and *Negative Log-Likelihood* (NLLLoss) for loss computation. It is also good to note that the pre-trained embeddings from the Filipino BERT are frozen throughout the training process. Figure 1 shows the system architecture in the context of Filipino Hate Speech.

4 RESULTS AND DISCUSSION

In the task of Hate Speech Detection in the generated TikTok Video Transcription dataset, experiments with Filipino BERT, LSTM with *Vocab2Index*, and other Machine Learning algorithms with a Term Frequency-Inverse Document Frequency (TF-IDF) encoding were implemented to generate a more comprehensive analysis on the generated dataset. This section discusses our analysis and interpretation of the generated results.

4.1 Implementation and Results Analysis

Experiments show that the best models from the data splitting setups come from the 60-20-20 split. Table 5 shows the performance of BERT for each class wherein 0 represents non-hate and 1 represents hate. As observed in Table 5, BERT classified poorly on predicting hate classes having a precision score of 50%, recall of 57%, and F1 score of 53%. In addition, BERT achieved a micro F1 score of 61%, macro F1 score of 60%, and a weighted average F1 score of 62% as shown in Table 5. The performance results of BERT are based on the nature of the dataset which provides a vague representation of hate speech at the same time provides very noisy and informal data for hate speech representation. To add, Filipino BERT does not generate a good generalization over the dataset. Also, the Twitter dataset used to pre-train the Filipino BERT does not quite fit the generated dataset since the nature of the Twitter data is different from the transcribed dataset since the transcription was derived from TikTok videos, which is a different platform with different contents. Moreover, Tagalog-English code-switching (Taglish) is observed throughout the dataset which could be the cause of confusion when classifying hate speech.

Table 4: Evaluation and Performance of the Models

	Micro F1	Macro F1	Weighted Average F1
BERT	0.61	0.60	0.62
LSTM	0.42	0.42	0.41
Bernoulli Naïve Bayes	0.74	0.72	0.74
Gaussian Naïve Bayes	0.62	0.60	0.62
Random Forest	0.73	0.64	0.69
Decision Tree	0.65	0.59	0.63
Linear SVC (SVM)	0.73	0.66	0.70
KNN Classifier	0.66	0.62	0.65

Table 5: BERT Class Performance

BERT	Precision	Recall	F1-score
non-hate	0.71	0.64	0.67
hate	0.50	0.57	0.53

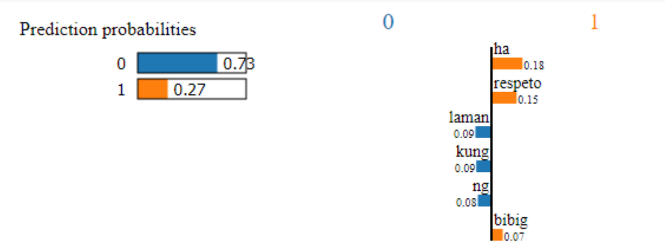


Figure 2: Prediction Probability of Bernoulli Naive Bayes for testing sample.

Furthermore, to have a much more comprehensive analysis, the researchers conducted other experiments on other machine learning approaches. Additional Deep Learning approach was implemented in the form of LSTM with Vocab2Index encoding wherein each word is assigned to a numerical value. When combined, forms a word encoding list which is then fed to the LSTM algorithm. To further extend the study analysis, other Machine Learning approaches were implemented: Bernoulli Naïve Bayes, Gaussian Naïve Bayes, Random Forest, Decision Tree, Linear Support Vector Classifier, and K Nearest Neighbor Classifier. Table 4 shows the performance results of the implemented algorithms.

Referring to Table 4 with our findings, BERT obtained the second to the lowest micro F1 score which is 61%. The lowest micro F1 score was obtained from the LSTM with a 42% micro F1 score. Meanwhile, machine learning algorithms results are as follows: Gaussian Naïve Bayes performed slightly better than BERT having a micro F1 score of 62%, macro F1 score of 60%, and a weighted average F1 score of 62%. Again, Decision Tree performed slightly better than Gaussian Naïve Bayes having a micro F1 score of 65%, macro F1 score of 59%, and a weighted average F1 score of 63%. Then, the KNN classifier performed slightly better than Decision Tree having a micro F1 score of 66%, macro F1 score of 62%, and a weighted average F1 score of 65%. Also, Random Forest and Linear SVC performed better than the previously mentioned machine learning algorithms, both algorithms are comparable to each other having a micro F1 score of 73%. Lastly, the highest micro F1 score obtained among all the models are 74% using Bernoulli Naïve Bayes. As observed in Table 4, machine learning algorithms perform better than the deep learning approaches due to the word occurrence in ML algorithms which better capture the class characteristics of the dataset when compared to DL approaches which provide the ability to capture the context of data.

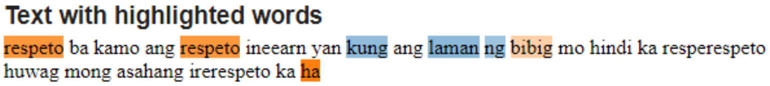
As shown in Table 4, Bernoulli NB performed best among all the other algorithms in classifying hate speech using the generated dataset. Table 6 shows the performance of Bernoulli NB for each class. It is observed that Bernoulli NB classifies better than BERT on predicting hate classes having a precision score of 68%, recall of 58%, and F1 score of 63%. In addition, Bernoulli NB achieved a 0.13 higher micro F1 score than BERT and 0.12 higher in macro and weighted F1 scores.

Based on the experiments conducted, Bernoulli NB is superior to other implemented algorithms, this is because the algorithm gives more accurate results due to its nature of performing best on short documents and small datasets, making it superior to the other models. Aside from its advantage on the dataset, the feature is treated independently and is converted into binary values. Bernoulli NB also explicitly gives a penalty to the model for non-occurrence of any feature instead of ignoring these features when compared to the multinomial variant of Naïve Bayes, which contributes to the performance of the prediction model. On the other hand, the generated BERT model wasn't able to get enough knowledge from the hate speech dataset used, due to the noisy, informal, and mixed-language environment from the transcribed dataset. In addition, BERT focuses on the underlying context of the dataset instead of the word occurrence, so when focusing on the dataset used, it performs poorly because some records include laughter and/or sarcasm and the matter of code-switching is observed in the dataset as well, compared with Bernoulli, this shows that classification is better because it focuses on the content itself, not just the context of the words.

To highlight the word occurrence feature used in Bernoulli NB, Figure 2 and Figure 3 shows one of some incorrect classifications of the Bernoulli Naïve Bayes prediction model, for this specific instance, this record is classified by the model as 0 (non-hate), however, is manually annotated by annotators as 1 (hate). Using the LIME library a generated visualization of which words contribute

Table 6: Bernoulli Naïve Bayes Class Performance

Bernoulli Naïve Bayes	Precision	Recall	F1-score
non-hate	0.77	0.84	0.80
hate	0.68	0.58	0.63



Text with highlighted words

respeto ba kamo ang respeto ineearn yan kung ang laman ng bibig mo hindi ka resperespeto huwag mong asahang irerespeto ka ha

Figure 3: Words that contribute to the classification of the testing sample

to determining the class of a specific text. Words such as “kung”, “laman”, “ng” are the highlighted words of which contribute to the prediction of the model towards “0” class (non-hate), while words such as “respeto” and “ha” are words that contribute to to the prediction of the model towards “1” class (hate). In the Filipino-Tagalog context, the word “respeto” (respect) can be interchangeably used depending on the intention of the context, to give an example: “Ako ay may respeto sa iyo” (I have respect for you) is a sentence with non-hate intention, however when the word is used aggressively such as “Wala kang respeto!” (You have no respect!), this sentence induces a negative and aggressive emotion when omitted. For the context of our dataset, the model detected that the word “respeto” is often used in a negative context. The word “ha” is simply an expression of asking what is said or asked, similar to the word “what”, other times it is used to express doubt, similar to the context of “what now?” when used aggressively. For this instance, the words: “kung”, “laman”, and “ng” when viewed alone and without context, are simply daily, conversational words that omit neutral emotions.

5 CONCLUSION AND FUTURE WORK

Hate speech detection in low resource languages such as Filipino/Tagalog pushes forward the current state of the art of research and technology of the language and contributes to the problem of machines understanding human language. In pursuit of the goal of the study, several experiments were conducted. Experiment shows that with a 61% Micro F1 score using the Filipino BERT for the hate speech detection with the TikTok Video Transcription dataset, BERT found it difficult to generalize and understand the underlying context of the data in the data prediction process, wherein Bernoulli Naïve Bayes generated a 74% Micro F1 Score, 13% better than BERT. Bernoulli Naïve Bayes was able to better determine the classes of the dataset due to the focus on word occurrence instead of context, at the same time, the algorithm provides better performance since the algorithm is tailored for short document datasets and generates binary features for binary classification. For this dataset, it is evident that the importance of word occurrence serves as a better basis for prediction over context. It is also good to note that the dataset generated only has 1000 records, thus the study can be further improved, and more findings and interpretations can be generated.

For future work, several tasks can be performed such as add more data to the dataset to allow prediction models to have a better understanding of the text. Also, include semantic, syntactic, and lexical features. Moreover, since text transcriptions were used, adding audio and video features can be explored to further boost the class representations. Lastly, explore ensemble learning algorithms or approaches (i.e., a voting ensemble of BERT, LSTM, and Bernoulli Naïve Bayes).

REFERENCES

[1] Allan, R. (2017, June 27). Hard Questions: Who Should Decide What Is Hate Speech in an Online Global Community? About Facebook. https://about.fb.com/news/2017/06/hard-questions-hate-speech/.

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[3] Mozafari, M., Farahbakhsh, R., & Crespi, N. (2019, December). A BERT-based transfer learning approach for hate speech detection in online social media. In International Conference on Complex Networks and Their Applications (pp. 928-940). Springer, Cham.

[4] Alatawi, H. S., Alhothali, A. M., & Moria, K. M. (2020). Detecting White Supremacist Hate Speech using Domain Specific Word Embedding with Deep Learning and BERT. arXiv preprint arXiv:2010.00357.

[5] Cruz, J. C. B., Resabal, J. K., Lin, J., Velasco, D. J., & Cheng, C. (2021, May 20). Exploiting News Article Structure for Automatic Corpus Generation. arXiv.org. https://arxiv.org/abs/2010.11574.

[6] Cruz, J. C. B., & Cheng, C. (2020). Establishing baselines for text classification in low-resource languages. arXiv preprint arXiv:2005.02068.

[7] Cruz, J. C. B., & Cheng, C. (2019). Evaluating language model finetuning techniques for low-resource languages. arXiv preprint arXiv:1907.00409.

[8] Manolescu, M., Löfflad, D., Saber, A. N. M., & Tari, M. M. (2019, June). TuEval at SemEval-2019 Task 5: LSTM Approach to Hate Speech Detection in English and Spanish. In Proceedings of the 13th International Workshop on Semantic Evaluation (pp. 498-502).

[9] Balabantaray, R. C., Mohammad, M., & Sharma, N. (2012). Multi-class twitter emotion classification: A new approach. International Journal of Applied Information Systems, 4(1), 48-53.

[10] Uyheng, J., & Carley, K. M. (2021). Characterizing network dynamics of online hate communities around the COVID-19 pandemic. Applied Network Science, 6(1), 1-21.

[11] Abro, S., Sarang Shaikh, Z. A., Khan, S., Mujtaba, G., & Khand, Z. H. Automatic Hate Speech Detection using Machine Learning: A Comparative Study. Machine Learning, 10, 6.

[12] Gaydhani, A., Doma, V., Kendre, S., & Bhagwat, L. (2018). Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. arXiv preprint arXiv:1809.08651.

[13] Malmasi, S., & Zampieri, M. (2017). Detecting hate speech in social media. arXiv preprint arXiv:1712.06427.

[14] Cabasag, N. V., Chan, V. R., Lim, S. C., Gonzales, M. E., & Cheng, C. (2019). Hate speech in philippine election-related tweets: Automatic detection and classification using natural language processing. Philippine Computing Journal, XIV No, 1.

[15] Adlaon, K. M. M., & Marcos, N. (2018, November). Neural Machine Translation for Cebuano to Tagalog with Subword Unit Translation. In 2018 International Conference on Asian Language Processing (IALP) (pp. 328-333). IEEE.

[16] Waseem, Z., & Hovy, D. (2016, June). Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In Proceedings of the NAACL student research workshop (pp. 88-93).

[17] Velasco, D. J. (2020). Pagsusuri ng RNN-based Transfer Learning Technique sa Low-Resource Language. arXiv preprint arXiv:2010.06447.

[18] Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017, May). Automated hate speech detection and the problem of offensive language. In Proceedings of the International AAAI Conference on Web and Social Media (Vol. 11, No. 1).

[19] Buitinck, L., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., & Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In G. Louppe (Ed.), ECML PKDD Workshop: Languages for Data Mining and Machine Learning (pp. 108–122).

[20] David, T. TikTok-Api: The Unofficial TikTok API Wrapper in Python. GitHub. https://github.com/davidteather/TikTok-Api/.

[21] *JDowloader*. JDownloader.org – Official Homepage. (2018, May 24) https://jdownloader.org/.

[22] Bird, S., Klein, E., Loper, E., (2019). Natural language processing with Python: analyzing text with the natural language toolkit. O'Reilly Media, Inc.