A Sentiment Analysis of Hate Speech in Philippine Election-Related Posts Using BERT Combined with Convolutional Neural Networks and Model Variations Incorporating Hashtags and ALL-CAPS

Micah Collette O. Mendoza
Department of Computer Science
College of Information and Computing Sciences
University of Santo Tomas, Manila, Philippines
collettemendoza.28@gmail.com

Mark Gabriel E. Ortiz
Department of Computer Science
College of Information and Computing Sciences
University of Santo Tomas, Manila, Philippines
markgabrielortiz7@gmail.com

Wayne Gabriel S. Nadurata
Department of Computer Science
College of Information and Computing Sciences
University of Santo Tomas, Manila, Philippines
wgn0914@gmail.com

Joshua Mari L. Padlan
Department of Computer Science
College of Information and Computing Sciences
University of Santo Tomas, Manila, Philippines
jmpadlan98@gmail.com

Charmaine S. Ponay
Department of Computer Science
College of Information and Computing Sciences
University of Santo Tomas, Manila, Philippines
csponay@ust.edu.ph

Abstract*—As the number of people who use X continually increases, the same thing is true for hate speech. A pressing need exists for automatic detection of posts that promote hate speech. The datasets gathered and validated from the base study by Argañosa et al. (2022) were used to categorize posts as either hate or non-hate and classify as either positive, negative or neutral with the utilization of Conventional Neural Networks. The partitioning of the labeled data into training and testing sets adhered to a ratio scheme: 70%-30%, 80%-20%, and 90%-10%. The model of this study, BERT-CNN, had an overall better performance in comparison to the base study, fastText CNN. Particularly, among the three splits, the BERT-CNN model for binary classification without the features of Hashtags and ALL-CAPS with the 90:10 split achieved the best performance with an accuracy of 93.55%, precision of 93.59%, and F1-score of 93.55%. For multi label classification, the BERT-CNN model demonstrated its optimal performance when incorporating hashtags, specifically with the 90:10 split, achieving an accuracy of 69.14%, precision of 68.44%, recall of 68.40%, and an F1-score of 67.41%. The innovative use of BERT word embeddings paired with CNN proved to excel in classifying Philippine election-related posts as hate or non-hate.

CCS Concepts

• Computing methodologies~Natural Language Processing~Sentiment Analysis~Model Misclassification~Hate Speech Detection • Computing methodologies~Convolutional Neural Networks~Transformer Networks~Text processing • Applied computing~Media Studies~Social media analytics

Keywords: *X, Hate Speech, Machine Learning, Deep Learning, Natural Language Processing, Sentiment Analysis, Convolutional Neural Networks, Bidirectional Encoder Representations from Transformers, Hashtags, ALL-CAPS*

I. INTRODUCTION

The term political hate speech is commonly referred to as an act of marginalizing or dehumanizing groups or individuals for their political ideology, beliefs or affiliation. This involves the use of coded phrasing with underlying discriminatory intent, persuasion to boycott, 'cancel' and even dissemination of false or misleading information. This act can be done by anyone at any time with more frequent action during elections or other political campaigns. These certain hate speeches are often engaged on various platforms such as social media which are being exploited to spread discrimination and misinformation [1].

Relevantly, X, formerly known as Twitter, served as a great platform to express filtered and unfiltered thoughts and ideas of both facts and opinion. In line with this, X became the netizen's choice for voicing political opinions, criticisms, and perceptions. This means that posts, formerly known as tweets, composed of hate speeches and offensive posts towards an opposing political faction or party are often visible and may lead to negative consequences which is a considerable problem as this may turn into threats or harassment that may generate fear within the receiving end. In fact, posts composed of hate speeches were almost at an all time high during the 2022

Philippine Election period due to its polar nature which caused a massive divide. During this time, supporters of a particular candidate would berate each other, call each other names, belittle the opposing candidate, or even much worse.

With the voluminous data available, it is a challenge to detect the amount of hate speech automatically. Computational models have been created to automatically detect hate speech such as the study from the University of Santo Tomas [2] with the detection of Filipino election-related posts with fastText CNN. However, despite the high accuracy of roughly 83%, there is still room for improvement. As such, the researchers aimed to answer the following questions with respect to the problem of the study: Can the combination of BERT with the CNN model improve the performance of the existing model proposed by the authors of the base study in detecting hate speech in Philippine election-related posts? To what extent can the model reach its optimal performance? How can this be improved? What insights can be derived from the analyzed text data with the utilization of BERT embeddings?

II. Related Literature

A. Natural Language Processing (NLP)

In computer science and artificial intelligence (AI), an area of research that is referred to as natural language processing, focuses on how to process natural languages like English [3]. It aims to develop new computational abilities that revolve on human language, such as information extraction from texts, language translation, question-answering, conversational interaction, and so forth. Although fundamental linguistic knowledge may be necessary for completing these various activities, success is ultimately determined by whether or not the work is completed. The task of automatically evaluating whether a text unit, such as a sentence or document, reflects a subjective or objective opinion or viewpoint is known as subjectivity classification in NLP. While objective text units communicate factual information or describe occurrences without expressing any personal viewpoints, subjective text units express personal sentiments, ideas, or beliefs [4].

B. Sentiment Analysis

The field of study known as sentiment analysis, sometimes known as opinion mining, examines people's opinions, sentiments, assessments, attitudes, and emotions about entities and their attributes as they are expressed in written text. The entities can be things like goods, services, businesses, people, occasions, problems, or subjects. The area of the problem represented by the field is enormous [5]. In another study, [6] conducted a proposal that analyzes the process of BERT word embeddings with the recommended hashtag feature using neural networking. The prediction results show that hashtags, especially hashtags clustering, is recommended for predictions and finding semantic similarity. It was also worth noting that the hashtags used in BERT word embeddings and other approaches such as Emhash improve output performance, and may also hinder prediction with its number of varieties. This makes hashtags in NLP models both a weakness and strength point.

In the study conducted by [7], an analysis of special orthographic characteristics which filter through in social mea platforms such as Twitter or X. This includes the characteristic of capitalization positioning in text. Results show that text with capitalization on Twitter or X shows a pattern which were labeled as meaningful capitalization, which is when capitalization is used on text with intent and expressive value, rather than uses for convenience such as abbreviations. In their study, meaningful capitalization in particular did hold an impact on sentiment analysis and functions as a conversational cue to clarify underlying meaning over text-based style communication.

C. Convolutional Neural Networks (CNN)

The Convolutional Neural Network (CNN) was utilized in the sentiment analysis on English posts related to the "Turkey Crisis 2018" topic. This was conducted in the study [8]. The sentiment analysis process began with data retrieval, followed by data classification using TextBlob to categorize posts as having positive, negative, or neutral sentiment. After undergoing training and evaluation, the CNN classifier model achieved an accuracy rate of 89%. During the test data testing process, the model achieved an accuracy rate of 88%. These accuracy results were then compared to those of the Naïve Bayes classifier model, which had an accuracy rate of 78%. It can be concluded that the classifier model utilizing a deep learning algorithm performs better in sentiment analysis than the NBC classifier model.

D. Bidirectional Encoder Representations from Transformers (BERT)

The widespread use of social media has led to the availability of a massive amount of data created by users, which can be analyzed to determine their emotions and opinions [9]. Profanities can also be used in this manner as a form of an aggressive or non-abusive manner, depending on the situation [10]. Several studies have shown the effectiveness of BERT in NLP tasks. However, the lack of publicly available Filipino post datasets regarding fire reports on social media has hindered the development of classification models for Filipino posts. Only a few insights were found, such as the discovery that the BERT model can also be useful in detecting and censoring Tagalog profanity in text media content [11]. To address this, a study was conducted by [12] which aimed to design and implement a system that can classify Filipino posts using different pre-trained BERT models. Using a dataset of 2,081 posts that contain fire-related posts, the authors created a model that can exclusively organize Filipino posts and compared the accuracy of different fine-tuned pre-trained BERT models. The results indicate a significant difference in accuracy among the pre-trained BERT models. The BERT Base Uncased WWM model performed the best, achieving a test accuracy of 87.50% and a train loss of 0.06 during training of the dataset, while the BERT Base Cased WWM model was the least accurate, with a test accuracy of 76.34% and a train loss of 0.2. These results suggest that the BERT Base Uncased WWM model is a reliable model for classifying fire-related posts in Filipino. However, the accuracy of the model may vary depending on the size of the dataset.

E. BERT-CNN

Studies on NLP have also found that combinations of BERT and CNN models have been proven to be more effective than using either one or alone [13]. A study by [14] highlighted the importance of sentiment analysis in improving the quality of commodities and influencing the purchasing decisions of consumers. However, the accuracy of existing sentiment analysis models needed to be improved. Therefore, the authors proposed a BERT-CNN model that improves the accuracy of sentiment analysis in commodity reviews. The results were compared to the Logistic Regression (LogReg) model used by [15]. The first CNN model used random word vectors and showed significant improvements in precision over the LogReg model, but lower recall. The second model used word2vec embeddings and improved recall by 7.3% compared to the random vector model, achieving an F-score of 78.29%. The third and fourth models used character n-grams in addition to word embeddings. The third model used only character n-grams as feature embeddings,

while the fourth model combined word2vec embeddings and character n-grams. The fourth model achieved the best precision, but the word2vec model without character n-grams had the best overall performance with precision, recall, and F-score values of 85.66%, 72.14%, and 78.29%, respectively. The CNN models outperformed the LogReg model in terms of precision and F1 score, while the LogReg model had better recall than the neural network models. In another study by [16], the authors combined pre-trained BERT and CNN models for text analysis. The study emphasized the importance of using pre-trained language models for downstream tasks, such as identifying offensive language. The results obtained from using only the BERT model were compared to those obtained by combining BERT with CNN, and the latter approach was found to be superior.

III. SIMULATIONS AND RESULTS

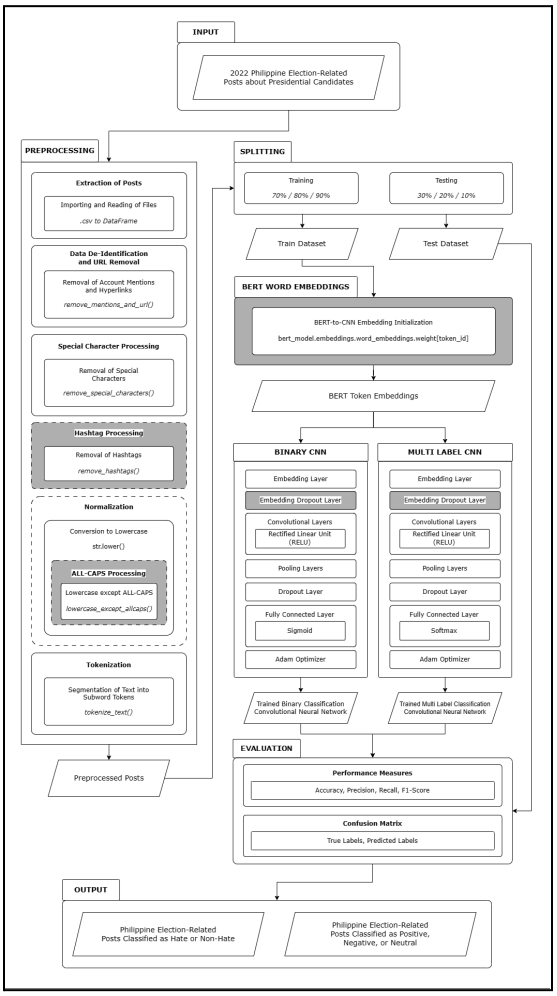

Figure 1. System Architecture

A. Hypotheses and Assumptions

The main goal of this study was to implement a model which detected and identified hate speech in Philippine election-related posts using the BERT-CNN model to improve the performance of the original fastText CNN model of the study conducted by [2]. The researchers of the study have deduced the following hypotheses:

Ho: There is no significant difference in the performance between the BERT-CNN model and fastText CNN in the sentiment analysis of Philippine election-related posts into hate or non-hate classification.

Ha: There is a significant improvement in the performance between the BERT-CNN model and fastText CNN in the sentiment analysis of Philippine election-related posts into hate or non-hate classification.

The following assumptions were assumed true in this study:

- The labeled data used in the sentiment analysis was correct and accurate.
- The dataset contained sufficient data for the system to output correct predictions.
- The size input for training and testing of data was sufficient to form an accurate analysis of the BERT-CNN model's performance.

B. Data Gathering and Preprocessing

The input data of the system architecture were extracted from the labeled dataset for binary and multi label classification of the base study [2] containing the gathered data of election-related posts from the X API during the 2022 Philippine elections. The validated dataset of the base study was the key comparison to the four types of dataset for each classification that resulted from the implemented BERT-CNN models: (A) BERT-CNN, (B) BERT-CNN with Hashtags, (C) BERT-CNN with ALL-CAPS, and (D) BERT-CNN with Hashtags and ALL-CAPS. Once the data was extracted, the preprocessing stage required several steps of cleaning to ensure that the model was to be fed with accurate and complete information. The data was then checked for any account mentions, URLs, special characters including emojis, diacritics, and numerics. In contrast to the methodology employed by the base study, where English and Filipino stopwords were eliminated, this study chose to retain them. The utilization of hashtags and ALL-CAPS were also one of the few changes the researchers revised from the base study's system. This created the eight datasets. For hashtag processing, hashtags were removed in the base BERT-CNN and BERT-CNN with ALL-CAPS models but retained for others utilizing hashtags. Concerning ALL-CAPS processing, this step occurred during normalization. Although every text was converted to lowercase, the casing for ALL-CAPS was preserved for models that utilized them. Finally, the textual data underwent tokenization using BERT tokenizer to facilitate subsequent vectorization processes.

C. Training and Testing

Following tokenization, the preprocessed data was split into training and testing datasets. During training, accurate classification across hate and non-hate labels, as well as positive, neutral, and negative sentiments took place. Adjustments to the training set's parameters were implemented, enhancing the model's capacity to discern underlying patterns and critical correlations within the data.

Once the training phase was complete, attention shifted to the testing set, comprising new, unseen data for the model to classify. The model's performance was then assessed based on what it had learned during training. The partitioning of the data into training and testing sets adhered to a ratio scheme: 70%-30%, 80%-20%, and 90%-10%, summing up to 100%. This strategic partitioning allowed for a robust evaluation of the model's generalization capabilities across various proportions of training and testing data.

D. Results

Table 1. Comparison of Performance Measures for fastText CNN and BERT-CNN models for Binary Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN
70:30	**90.30 %**	85.61 %	**90.30 %**	85.97 %	**90.30 %**	85.7 0%	**90.30 %**	85.59 %
80:20	**90.62 %**	85.94 %	**90.64 %**	86.02 %	**90.63 %**	85.9 2%	**90.62 %**	85.92 %
90:10	**93.55 %**	86.52 %	**93.59 %**	86.85 %	**93.58 %**	86.4 7%	**93.55 %**	86.49 %

Table 2. Comparison of Performance Measures for fastText CNN and BERT-CNN models for Multi Label Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN	BERT-CNN	fastText CNN
70:30	**65.54 %**	62.63 %	**65.46 %**	61.9 7%	**65.5 2%**	62.5 4%	**63.21 %**	61.82 %
80:20	**66.28 %**	62.89 %	**65.82 %**	62.1 1%	**66.2 4%**	62.7 9%	**64.49 %**	62.07 %
90:10	**67.84 %**	63.80 %	**66.60 %**	63.0 9%	**67.1 0%**	63.4 1%	**66.27 %**	62.74 %

The comparison between the BERT-CNN and fastText CNN models for binary classification revealed notable differences in performance across various train-test splits. BERT-CNN consistently outperformed fastText CNN in terms of accuracy, precision, recall, and F1-score. In the 70:30 split, BERT-CNN achieved an accuracy of 90.30%, while fastText CNN lags behind at 85.61%. This trend continued across splits, with BERT-CNN maintaining higher precision and recall values. The 90:10 split showcased a substantial improvement in BERT-CNN's performance, with an accuracy of 93.55%, precision of 93.59%, recall of 93.58%, and F1-score of 93.55%. In contrast, fastText CNN exhibited more modest gains as the training set size increased.

Overall, BERT-CNN demonstrated a clear advantage over fastText CNN, highlighting its superior predictive capabilities and suitability for binary classification tasks across different train-test splits. The BERT-CNN model for multi label classification also exhibited a notable superiority in accuracy, surpassing the fastText CNN model by 2.91% absolute accuracy on the 70:30 train-test split, 3.39% on the 80:20 train-test split, and 4.01% on the 90:10 train-test split. These findings suggest that the BERT-CNN model outperforms the fastText CNN model across different data split ratios, emphasizing its effectiveness in the context of the study.

Table 3. Comparison of Performance Measures for BERT-CNN with Hashtags and BERT-CNN without Hashtags models for Binary Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ HT	fBERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT
70:30	**90.69 %**	90.30 %	**90.69 %**	90.30 %	**90.69 %**	90.30 %	**90.69 %**	90.30 %
80:20	90.14 %	**90.62 %**	90.16 %	**90.64 %**	90.15 %	**90.63 %**	90.14 %	**90.62 %**
90:10	92.38 %	**93.55 %**	91.76 %	**93.59 %**	92.86 %	**93.58 %**	92.31 %	**93.55 %**

Table 4. Comparison of Performance Measures for BERT-CNN with Hashtags and BERT-CNN without Hashtags models for Multi Label Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ HT	fBERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT	BERT-CNN w/ HT	BERT-CNN w/o HT
70:30	**67.93 %**	65.54 %	**67.37 %**	65.46 %	**67.90 %**	65.52 %	**66.90 %**	63.21 %
80:20	**67.06 %**	66.28 %	**67.12 %**	65.82 %	**67.09 %**	66.24 %	**65.30 %**	64.49 %
90:10	**69.14 %**	67.84 %	**68.44 %**	66.60 %	**68.40 %**	67.10 %	**67.41 %**	66.27 %

The comparison between BERT-CNN models with and without hashtags (HT) for binary classification highlighted subtle variations in their performance metrics. In the 70:30 split, the BERT-CNN with HT achieved an accuracy of 90.69%, precision of 90.69%, recall of 90.33%, and an F1-score of 90.51%, surpassing the corresponding metrics of the model without HT by a small margin. However, BERT-CNN without HT consistently exhibited slightly higher accuracy, precision, recall, and F1-score values in the 80:20 and 90:10 splits. This trend showcased a consistent advantage for the model without HT. While both models demonstrated strong binary classification capabilities, the absence of hashtags appeared to contribute to a modest but consistent improvement in performance metrics for BERT-CNN. The BERT-CNN model for multi label classification with HT achieved the highest absolute accuracy, surpassing the model without HT by 1.23% on the 70:30 split ratio, 0.78% on the 80:20 split, and 0.33% on the 90:10 split. These results highlighted the positive impact of incorporating hashtags in the multi label classification task, indicating that the inclusion of this contextual information contributed to the model's ability to accurately classify sentiments. The consistent outperformance of the BERT-CNN model with HT, especially in terms of accuracy, reinforces the relevance and effectiveness of leveraging hashtag information for improved sentiment analysis in multi label scenarios.

Table 5. Comparison of Performance Measures for BERT-CNN with Hashtags and BERT-CNN without Hashtags models for Binary Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)
70:30	89.58 %	**90.30 %**	88.49 %	**90.30 %**	**90.60 %**	90.30 %	89.53 %	**90.30 %**
80:20	88.48 %	**90.62 %**	87.64 %	**90.64 %**	89.37 %	**90.63 %**	88.50 %	**90.62 %**
90:10	91.41 %	**93.55 %**	90.00 %	**93.59 %**	92.86 %	**93.58 %**	91.41 %	**93.55 %**

Table 6. Comparison of Performance Measures for BERT-CNN with ALL-CAPS and BERT-CNN without ALL-CAPS models for Multi Label Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)
70:30	64.71 %	**65.54 %**	64.05 %	**65.46 %**	64.65 %	**65.52 %**	**63.42 %**	63.21 %
80:20	65.04 %	**66.28 %**	64.08 %	**65.82 %**	64.89 %	**66.24 %**	63.95 %	**64.49 %**
90:10	67.71 %	**67.84 %**	**67.56 %**	66.60 %	**67.23 %**	67.10 %	65.61 %	**66.27 %**

The comparison of performance measures for BERT-CNN with and without ALL-CAPS (AC) in binary classification revealed notable distinctions. The uncased BERT-CNN, without AC, consistently outperformed its cased counterpart with AC across all splits, demonstrating superior accuracy, precision, recall, and F1-score—except for the 70:30 split, where BERT-CNN with AC exhibited higher recall at 90.60%. Specifically, at the 90:10 split, the uncased model achieved an impressive accuracy of 93.55%, surpassing the cased model's 91.41%. This suggests that the uncased BERT-CNN model exhibited greater proficiency in handling the classification task. The BERT-CNN model for multi label classification with AC demonstrated specific improvements. Specifically, BERT-CNN with AC outperformed the model without AC by 0.21% in F1-Score on the 70:30 split ratio. Additionally, there were gains of 0.96% in precision and 0.13% in recall on the 90:10 split for the model with AC. These findings suggest that while the uncased model without AC showed a slight edge in overall performance, the inclusion of AC contributed to specific enhancements in precision, recall, and F1-Score, particularly in certain split ratios.

Table 7. Comparison of Performance Measures for BERT-CNN with Hashtags and ALL-CAPS and BERT-CNN without Hashtags and ALL-CAPS models for Binary Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)	BERT-CNN w/ AC (Cased)	BERT-CNN w/o AC (Uncased)
70:30	89.58 %	**90.30 %**	88.49 %	**90.30 %**	**90.60 %**	90.30 %	89.53 %	**90.30 %**
80:20	88.48 %	**90.62 %**	87.64 %	**90.64 %**	89.37 %	**90.63 %**	88.50 %	**90.62 %**
90:10	91.41 %	**93.55 %**	90.00 %	**93.59 %**	92.86 %	**93.58 %**	91.41 %	**93.55 %**

Table 8. Comparison of Performance Measures for BERT-CNN with Hashtags and ALL-CAPS and BERT-CNN without Hashtags and ALL-CAPS models for Multi Label Classification

Split	Accuracy		Precision		Recall		F1-Score	
	BERT-CNN w/ HT and AC (Cased)	BERT-CNN w/o HT and AC (Uncased)	BERT-CNN w/ HT and AC (Cased)	BERT-CNN w/o HT and AC (Uncased)	BERT-CNN w/ HT and AC (Cased)	BERT-CNN w/o HT and AC (Uncased)	BERT-CNN w/ HT and AC (Cased)	BERT-CNN w/o HT and AC (Uncased)
70:30	**66.45 %**	65.54 %	**65.98 %**	65.46 %	**66.36 %**	65.52 %	**65.49 %**	63.21 %
80:20	65.89 %	**66.28 %**	65.15 %	**65.82 %**	65.81 %	**66.24 %**	**64.96 %**	64.49 %
90:10	65.10 %	**67.84 %**	65.37 %	**66.60 %**	64.38 %	**67.10 %**	61.74 %	**66.27 %**

The uncased BERT-CNN without HT and AC for binary classification consistently outperformed its cased counterpart across all splits, exhibiting higher accuracy, precision, recall, and F1-score—except for the 80:20 split, where cased BERT-CNN with HT+AC exhibited higher accuracy, precision and F1-score, and for the 70:30 split with higher recall. The performance of the cased model, while competitive, showed less variability across different splits, with minimal improvements in precision and recall. The uncased BERT-CNN without HT and AC for multi label demonstrated superior overall performance compared to its cased counterpart with HT and AC. While the cased model exhibited superiority in the 70:30 split across all evaluation metrics, including a notable improvement in F1-score in the subsequent split, the overall trend showed a decline and inconsistency in other metrics.

Table 9. McNemar's Test with Holm-Bonferroni for BERT-CNN and fastText CNN models for Binary Label Classification

Split	Hypothesis Testing (α = 0.05)		
	p-value	HB correction	Result
70:30	0.0000001871	0.016	Reject Null Hypothesis
80:20	0.0000537307	0.05	Reject Null Hypothesis
90:10	0.0000048175	0.025	Reject Null Hypothesis

Table 10. McNemar's Test with Holm-Bonferroni for BERT-CNN and fastText CNN models for Multi Label Classification

Split	Hypothesis Testing (α = 0.05)		
	p-value	HB correction	Result
70:30	0.0064897136	0.016	Reject Null Hypothesis
80:20	0.0118929794	0.025	Reject Null Hypothesis
90:10	0.0386433487	0.05	Reject Null Hypothesis

Based on the results of a McNemar's test with Holm-Bonferroni correction for BERT-CNN and fastText CNN models for both binary and multi label classification, the BERT-CNN model outperformed the fastText CNN model for binary classification with the binary labeled dataset as well as the multi classification with the multi labeled dataset. The results on Table 9 and Table 10 have shown that there is a significant improvement in the performance between the BERT-CNN and fastText CNN models. This suggests that BERT can exceed fastText as a powerful language model to extract meaningful features as well as underlying context from posts.

Table 11. Sample Candidate Names with Prediction Values

Candidate Names	Prediction Value	Label
BBM	0.38	Non-Hate
bongbong	**0.91**	**Hate**
Marcos	**0.68**	**Hate**
Leni	0.19	Non-Hate
Robredo	0.19	Non-Hate

The BERT-CNN models heavily relied on contextual data to discern specific words or phrases as conversational cues, embedding significant value for distinguishing between hate and non-hate posts. The integration of BERT embeddings into the BERT-CNN model played a pivotal role, particularly in the context of sentiment analysis for Philippine election-related posts. Notably, the inclusion of candidate names significantly influenced the model's predictions, as these names carried substantial word embedding values derived from the dataset. The study, centered around the latest Philippine presidential elections, highlighted the prevalence of candidate names in posts. Table 10 presented candidate names alongside their corresponding prediction values derived from the BERT-CNN with HT and AC model. It also provided insights into how the model assigned labels based on the presence of candidate names, revealing nuanced dynamics in hate speech detection within the election context.

The strong influence of candidate names on the prediction's classification label suggested their role as prominent conversational cues. For instance, Marcos was predominantly linked to posts labeled as hate. However, a closer examination of the dataset revealed a significant disparity in the number of hate-labeled posts containing the names of Bong Bong Marcos and Leni Robredo. This observation suggested that the large value associated with Marcos's name could be attributed to a higher number of labeled hate posts compared to non-hate posts. This was in contrast to posts related to Robredo, where both hate and non-hate posts were more evenly distributed. This showed that names, specifically those with relation to the presidential election, hold a large embedding value on the model's classification and emphasizing the need for careful consideration in the classification process. While the utilization of BERT word embeddings proved advantageous in enhancing the model's overall performance, it also brings a risk to misclassifying text based on the embeddings of a certain word, namely the names of the campaigned presidential candidates against the whole context.

IV. Conclusions and Recommendations

A. Conclusions

Having deployed the BERT-CNN model, this study achieved its primary goal of assessing the effectiveness of the BERT-CNN model in identifying political hate speech on X, demonstrating its proficiency in sentiment analysis of Filipino election-related posts. The comparison with the model proposed by [2] consistently favored BERT-CNN, validating its superior performance. The following conclusions were derived based on the deployment of the presented system and summary of findings:

The BERT-CNN model demonstrated exceptional proficiency in classifying Philippine election-related posts into hate or non-hate categories, achieving high accuracy, precision, recall, and F1-score, each exceeding 90%. The consistency of these outstanding results across a spectrum of train-test splits highlighted the model's robust performance. Regardless of the variations introduced, such as the incorporation of Hashtags, ALL-CAPS, or both, the BERT-CNN model maintained its superior classification capabilities. These findings underscored the model's resilience and effectiveness in handling diverse textual data related to Philippine elections, making it a reliable and versatile tool for sentiment analysis in this context.

The comparison between the BERT-CNN and fastText CNN models revealed notable advancements in hate speech detection achieved by the BERT-CNN model. Demonstrating consistent superiority across diverse performance measures and splits, the BERT-CNN model exhibits a robust ability to classify effectively in various scenarios. The key strength of BERT-CNN lies in its capacity to capture intricate contextual relationships within language, rendering it particularly well-suited for complex sentiment analysis tasks. The model's persistent outperformance implies its potential to

offer more accurate and nuanced predictions when compared to the fastText CNN. Furthermore, the results of the McNemar's Test with Holm-Bonferroni correction affirmed a significant improvement in performance for the BERT-CNN model compared to the fastText CNN in the sentiment analysis of election-related posts, specifically in hate or non-hate classification, across all train-test splits.

The integration of BERT embeddings into the BERT-CNN model, significantly enhanced the model's performance in sentiment analysis of Philippine election-related posts. The study delved into the intricate dynamics of hate speech detection within the context of the Philippine presidential elections. One noteworthy aspect is the substantial influence of candidate names on the model's predictions, indicating a large word embedding value. Candidate names emerged as robust conversational cues, becoming integral to the classification of posts as hate or non-hate. This is exemplified by the detailed analysis of candidate names such as Marcos and Leni Robredo, showcasing their varying frequencies in hate and non-hate posts. The BERT-CNN model demonstrated its ability to learn from training data, identifying frequent words like "bongbong" and "Marcos" as indicators for hate predictions, thereby improving its contextual understanding. However, the study also uncovered potential misclassifications, emphasizing the model's reliance on training data and the context-dependent nature of BERT embeddings. Despite the risk of misclassification, the BERT word embeddings approach was deemed advantageous in enhancing the model's overall performance, showcasing its ability to embed words based on contextual usage and providing valuable insights into the sentiment analysis of election-related posts.

B. Recommendations

Based on the findings of the study and its comprehensive conclusions, the researchers recommend the following:

Choice of Model or Further Hyperparameter Tuning: Although the created BERT-CNN model showed excellent results when it comes to Accuracy, Precision, Recall, and F1 score, it was noticed by the group that the CNN model tends to overfit early which is also noticeable in the base study. Hence, another recommendation is to use another model to be paired with BERT such as LSTM which is a more capable model in terms of natural language processing. At the same time, another option is to further experiment with the hyperparameters of CNN, especially the dropout and weight decay of the Adam optimizer which is set to 0.5 and 0.0 respectively which is similar to the base model. Alternatively, researchers can also modify the architecture of the CNN model as the architecture may be too complicated which led to early overfitting.

Additional Data for Model Training and Testing: The researchers recommend incorporating additional data to enrich the general context input for BERT and CNN algorithms, aiming to broaden the understanding of language patterns, sentiments, and contexts associated with hate speech. This approach, grounded in the idea of utilizing a diverse and extensive dataset, is expected to enhance model generalization across various scenarios and improve predictive capabilities. The inclusion of more data not only enhances model robustness but also enables effective performance on diverse inputs, particularly in dynamic and context-dependent language applications, such as those found in online social platforms.

BERT Embeddings as the Input of CNN: The researchers suggest exploring the use of BERT embeddings as the input for CNN models. BERT embeddings capture rich semantic information about words and their contextual relationships, and integrate them as input features for a CNN. Combining the strengths of both models may lead to a more comprehensive representation of text data, potentially improving the overall performance in hate speech detection tasks. This approach allows for flexibility and innovation in the choice of model architectures, fostering advancements in the field of natural language processing and hate speech detection.

Further Experiments with Hashtags: At the same time, the researchers also recommended experimenting with hashtags like removing the hashtag symbol “#” instead of removing the hashtags entirely as this study showed that BERT is capable of handling them but its embedded value is not as significant as the researchers hoped. However, the results of this might end up the same as the candidate names as it might tend to misclassify inputs because of heavy embeddings.

Extension to Other Domains: The extension of this study to evaluate the generalizability of the BERT-CNN model across diverse domains and contexts is vital. Assessing its performance outside the realm of Philippine election-related posts provides valuable insights into the model's versatility and adaptability. Such an extension could contribute to the development of more universally applicable hate speech detection systems.

By embracing these recommendations, future research endeavors can build upon the foundation laid by this study and advance the field of hate speech detection towards more nuanced, contextually aware, and adaptable models.

ACKNOWLEDGEMENT

The researchers wish to express heartfelt appreciation to the following individuals for their unwavering support, assistance, and significant contributions to the successful completion of this study.

Foremost, gratitude is extended to family and friends whose belief in the researchers and encouragement were constant sources of motivation throughout the entire process.

To Asst. Prof. Charmaine S. Ponay, their thesis adviser, whose time, patience, and invaluable advice greatly influenced the academic research. The imparted knowledge will forever remain an integral part of the study, and without her guidance, this paper would not have come to fruition.

To Asst. Prof. Cherry Rose Estabillo, their thesis coordinator, for providing the opportunity to undertake the study and for offering valuable recommendations and suggestions to overcome the challenges encountered.

To Asst. Prof. Janette E. Sideño, Assoc. Prof. Perla P. Cosme, and Mr. Von Guron, the thesis defense panelists, for their understanding and constructive feedback, which contributed to strengthening the integrity of the study.

Lastly, heartfelt appreciation is directed to the Almighty Father, whose blessings bestowed the researchers with life, wisdom, and the passion to see the research through to its conclusion.

REFERENCES

[1] Hate speech and incitement to hatred or violence. OHCHR. (n.d.). https://www.ohchr.org/en/special-procedures/sr-religion-or-belief/hate-speech-and-incitement-hatred-or-violence#:~:text=As%20a%20matter%20of%20principle,peaceful%2C%20inclusive%20and%20just%20societies.

[2] Argañosa, S. C. D., Marasigan, R. L., Villanueva, J. J. S., & Wenceslao, J. K. C. (2022). Hate Speech in Filipino Election-Related Tweets: A Sentiment Analysis Using Convolutional Neural Networks.

[3] Hapke, H., Howard, C., & Lane, H. (2019). Natural Language Processing in Action: Understanding, analyzing, and generating text with Python. Simon and Schuster.

[4] Eisenstein, J. (2019). Introduction to natural language processing. MIT press.

[5] Liu, B. (2020). Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge university press.

[6] Kaviani, M., & Rahmani, H. (2020). EmHash: Hashtag Recommendation using Neural Network based on BERT Embedding. Sci-hub. https://sci-hub.se/10.1109/icwr49608.2020.9122296

[7] Chan, S., & Fyshe, A. (2018). Social and emotional correlates of capitalization on Twitter. ACL Anthology. https://aclanthology.org/W18-1102/

[8] Sunarya, P. A., Refianti, R., Mutiara, A. B., & Octaviani, W. (2019). Comparison of accuracy between convolutional neural networks and Naïve Bayes Classifiers in sentiment analysis on Twitter. International Journal of Advanced Computer Science and Applications, 10(5).

[9] Chiorrini, A., Diamantini, C., Mircoli, A., & Potena, D. (2021). Emotion and sentiment analysis of tweets using BERT. In EDBT/ICDT Workshops (Vol. 3).

[10] Galinato, V., Amores, L., Magsino, G. B., & Sumawang, D. R. (2023). Context-Based Profanity Detection and Censorship Using Bidirectional Encoder Representations from Transformers. Available at SSRN 4341604.

[11] 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.

[12] Mingua, J., Padilla, D., & Celino, E. J. (2021, November). Classification of Fire Related Posts on Twitter Using Bidirectional Encoder Representations from Transformers (BERT). In 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM) (pp. 1-6). IEEE.

[13] Li, W., Gao, S., Zhou, H., Huang, Z., Zhang, K., & Li, W. (2019, December). The automatic text classification method based on BERT and feature union. In 2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS) (pp. 774-777). IEEE.

[14] Dong, J., He, F., Guo, Y., & Zhang, H. (2020, May). A commodity review sentiment analysis based on BERT-CNN model. In 2020 5th International conference on computer and communication systems (ICCCS) (pp. 143-147). IEEE.

[15] 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).

[16] Safaya, A., Abdullatif, M., & Yuret, D. (2020, December). Kuisail at semeval-2020 task 12: BERT-CNN for offensive speech identification in social media. In Proceedings of the Fourteenth Workshop on Semantic Evaluation (pp. 2054-2059).