Towards Generalizable Hate Speech Detectors, an Analysis of Model Bias and Ensemble Methods

By

Ricky Yu, BSc, University of Toronto St.George, 2023

A Major Research Paper

Presented to Toronto Metropolitan University

In partial fulfillment of the requirements for the degree of

Master of Science

In the Program of

Data Science and Analytics

Toronto, Ontario, Canada, 2024

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF AN MRP

I hereby declare that I am the sole author of this MRP. This is a true copy of the MRP, including any required final revisions.

I authorize Toronto Metropolitan University to lend this MRP to other institutions or individuals for the purpose of scholarly research.

I further authorize Toronto Metropolitan University to reproduce this MRP by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my MRP may be made electronically available to the public.

Towards Generalizable Hate Speech Classifiers, an Analysis of Model Bias and Ensemble Methods

Ricky Yu

Master of Science 2024

Data Science and Analytics

Toronto Metropolitan University

ABSTRACT

In this paper, we propose and evaluate 3 models that augment state-of-the-art hate speech classifier models in their ability to handle unseen data and dataset bias. We first performed exploratory data analysis to reveal the distribution of the training datasets to determine the degree of their bias towards certain topics. We then evaluate the performance of several state-of-the-art feature representation and deep learning models on the training datasets. The best performing model was then used to evaluate the effect of unseen data and bias on model performance. Finally, we test the effect of dataset augmentation as well as 2 ensemble methods for alleviating the impact of unseen data and dataset bias.

Keywords:

Hate Speech, Text Classification, Bias, Deep Learning, Ensemble

ACKNOWLEDGEMENTS

I would like to thank Professor Anatoliy Gruzd for his assistance and guidance in delivering this project. His insight and expertise in the field of social media analytics and research helped immensely in guiding the direction of my MRP. I am especially grateful for his consistent feedback and advice for the academic research process that allowed me to deliver novel research to the standard expected by this institution and academia.

Thank you, Professor Anatoliy

Table of Contents

List of Figures

List of Tables

Introduction

Background

The pervasive influence of social media platforms has led to an exponential increase in the volume of content shared online. This growth in online content and discourse is also accompanied by a small but significant portion of toxic and hateful content. As such, hate speech detection has emerged as a crucial component in analyzing and addressing the influence of toxicity and hate online.

Over the years, researchers have proposed various models to tackle hate speech detection. Recent advancements in deep learning have demonstrated promising results in enhancing the performance of hate speech detection models compared to their classical counterparts. However, state-of-the-art models have been shown to degrade performance when evaluated in generalized use cases. Thus, the development of a generalizable and accurate hate speech detector is an important but difficult task. Nevertheless, understanding and combating the influence of toxic and hateful content in modern and future social media platforms is immensely valuable, and an accurate and generalizable hate detector would greatly aid in this endeavor.

Research Question

To accomplish the task of improving generalizable hate speech detectors, it is necessary to determine the successes and failures of the current state-of-the-art models. The project will first evaluate classical and deep learning models from literature to break down their performance. The best performing model will then be used in cross-dataset evaluation to measure their performance. We will then seek to improve state-of-the-art performance in generalized tasks. To conclude the project, the resulting improved generalized detector will be used to identify hate speech in real-world data.

In summary, the research questions are:

A. How well do the state-of-the-art hate speech detectors perform in standard hate speech detection tasks, and how does different model architecture affect model performance?

B. How do the same state-of-the-art models perform when used in generalized use cases?

C. What modifications can be made to the state-of-the-art models to improve their performance in generalized use cases?

Literature Review

The literature review began with a deep dive into the research landscape of hate speech detection. Here, the goals were to understand what state-of-the-art methods were achieving the best performance. For this purpose, Yin & Zubiaga (2021) and Jahan & Oussalah (2023) revealed that recent advances in deep learning models i.e. CNNs and RNNs, had achieved greater performance than classical machine learning models such as logistic regression and SVM. Additionally, Yin & Zubiaga (2021) revealed that while the state-of-the-art models were reporting high-performance values in individual datasets, an attempt at applying trained models on new datasets resulted in subpar performance. Fortuna et al. (2021) would go further in their analysis, revealing the degrading performance of models when applied to new datasets was due to the dissimilarity between labels and out-of-vocabulary words, while Zhang & Luo (2018) focused on the poor performance of state-of-the-art models on the hate-class.

The next section of the literature review covers major model architecture from the literature. Davidson et al. (2017) is a prominent paper in the field which primarily tested an SVM model for hate speech detection. They found that SVM mostly succeeded in documents where keywords related to hate were featured but remarked that implicit hate speech could be entirely missed by the SVM model. In Jahan & Oussalah (2023) the authors listed SVM as the most popular classical machine learning model, followed by Logistic Regression and Naïve Bayes. To form a baseline of minimum performance, all 3 classical models will be evaluated across multiple datasets.

As stated in both Yin & Zubiaga (2021) and Jahan & Oussalah (2023), deep learning is quickly becoming the most popular model type in the field. The work done by Zhang et al. (2018) was a major step in model performance and proposed the CNN + RNN model. CNN (Convolutional Neural Network) layers allow the model to capture local semantic data, allowing it to analyze relations between nearby words, while the RNN (Recurrent Neural Network) takes the local semantic data from the CNN layers and captures the sequential features. Together, CNNs and RNNs are expected to capture contextual and implicit hate speech better than classical methods like SVM. The authors of Zhang et al. (2018) directly compared their model performance to the SVM model from Davidson et al. (2017) where it consistently outperformed in F1. This reflects what was reported by Yin & Zubiaga (2021) and Jahan & Oussalah (2023), highlighting the superiority of deep learning models compared to classical models.

The performance of the CNN + RNN model of Zhang et al. (2018) however was challenged by the 2 layer CNN model from Roy et al. (2020). The authors of Roy et al. (2020) found that their 2-layer CNN model outperformed their CNN + RNN model, which stands in opposition to the reported performance of CNN + RNN models shown in Zhang et al. (2018). Since both works tested on different datasets, this project will evaluate the best performing models from both literature on the same datasets to compare their performance.

A key component of NLP is the vectorized representation of the text passed into the main classifier. The authors of Ali et al. (2022) performed extensive tests involving both classical vectorization models i.e. TF-IDF, CountVectorizer, Word2Vec, FastText and transformer embedding models in the form of BERT, DistilBERT and XLMRoberta. Their analysis revealed that the transformer embeddings outperformed classical embeddings. This project will also evaluate TFI-DF and Word2Vec, as well as BERT, ALBERT and ELECTRA. Both ALBERT and ELECTRA offer faster compute time while performing similarly to BERT. If their performance is close enough to BERT, these two options could be reasonable substitutes for BERT.

The final major model architecture suggestion came from Mullah & Zainon (2021). This paper reviewed current and past research on hate speech detection and noted a lack of testing on ensemble methods. Mazari et al. (2024) comes in as a recent contribution to the field, featuring ensemble learning leveraging BERT and stacked RNN layers used in the ensemble. The results concluded that the ensemble methods outperformed the individual models, and notably reduced misclassification. The individual models tested in this paper were rudimentary, and this project will aim to implement ensemble methods using the more advanced models seen in Zhang et al. (2018) and Roy et al. (2020).

For model training, a collection of open-source datasets from the literature has been collected. Davidson et al. (2017) and Basile et al. (2019) provide datasets collected from Twitter and were gathered using a lexicon approach, which selected documents from Twitter that included terms commonly associated with hate. Mollas et al. (2022) was collected from YouTube and Reddit employing a filtering approach that first classified documents using a weak classifier, which was then passed to multiple human annotators for scrutiny and correction. Qian et al., (2019) was collected from Reddit and Gab, with both Qian et al. (2019) and Davidson et al. (2017) using the lexicon listed on HateBase.org for their lexicon based document selection. HateBase.org is a crowdsourced repository of hate speech terms intended for use by researchers and organizations to analyze hate speech. The final dataset, cjadams (2017), is provided by Wikipedia and was listed on Kaggle for a model training challenge. Together these five datasets provide a large sample of documents collected through various methods and domains.

Exploratory Data Analysis

This project will be training its models on 5 datasets, each collected using different methods from different sources, as listed in the previous section. For brevity, we will refer to the datasets provided from the literature as following:

- Davidson et al. (2017): **Davidson**
- Basile et al. (2019): **Hateval**
- Mollas et al. (2022)**: ETHOS**
- cjadams (2017)**: Jigsaw**
- Qian et al. (2019)**: Qian**

Except for Jigsaw, each dataset is text-based, with a binary label for whether the text is considered hate speech or not. Jigsaw is a multi-label dataset, with labels for various forms of toxicity, of which we will only select the identity attack label, which converts the dataset into a binary dataset as well.

All datasets were then preprocessed by removing non-alphabetical characters, and platform-specific tags like retweets and hashtags as well as hyperlinks. This was done using a combination of regex matches and replacements using Python.

With the datasets preprocessed, data analysis was performed on each dataset to better understand their content.

Dataset Size

Each dataset was collected using different APIs from different sources, which greatly affected the size of each dataset.

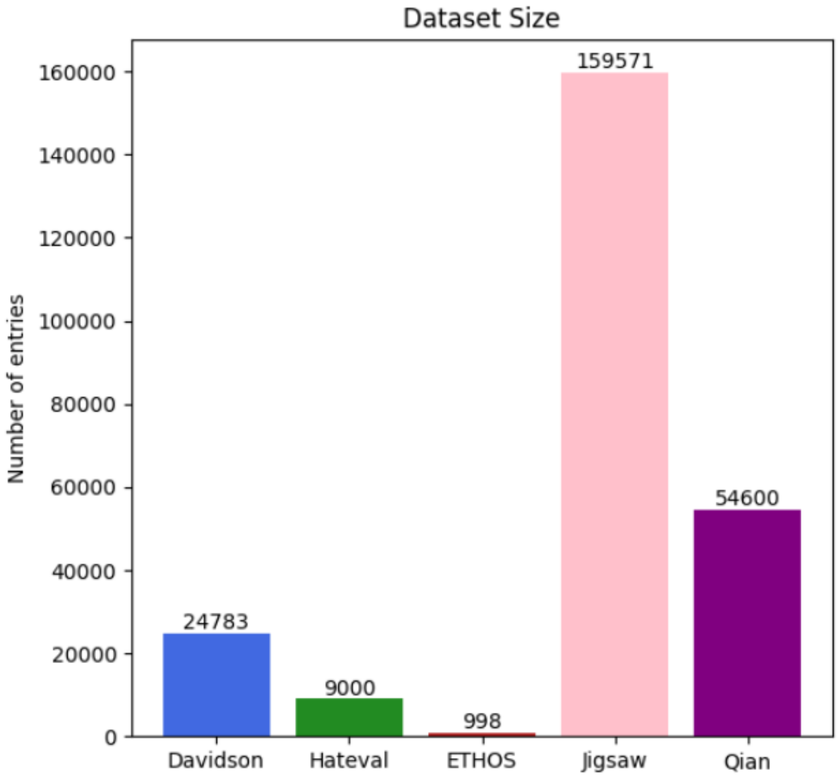

Figure 1: Dataset size comparison

Figure 1: Dataset size comparison

Most of the datasets feature a sizable number of documents (Figure 1), except for ETHOS. The authors of the ETHOS dataset Mollas et al. (2022) admit the dataset size is sub-par compared to others in the field, but are confident that their thorough selection pipeline produced a dataset that is fairly unbiased compared to others in the field.

Dataset Distribution

Beyond the size of the dataset, another important statistic is the distribution of hate--classified documents. Hate speech is only a small portion of the overall discourse on social media platforms, and this is expected to be reflected in the datasets as well.

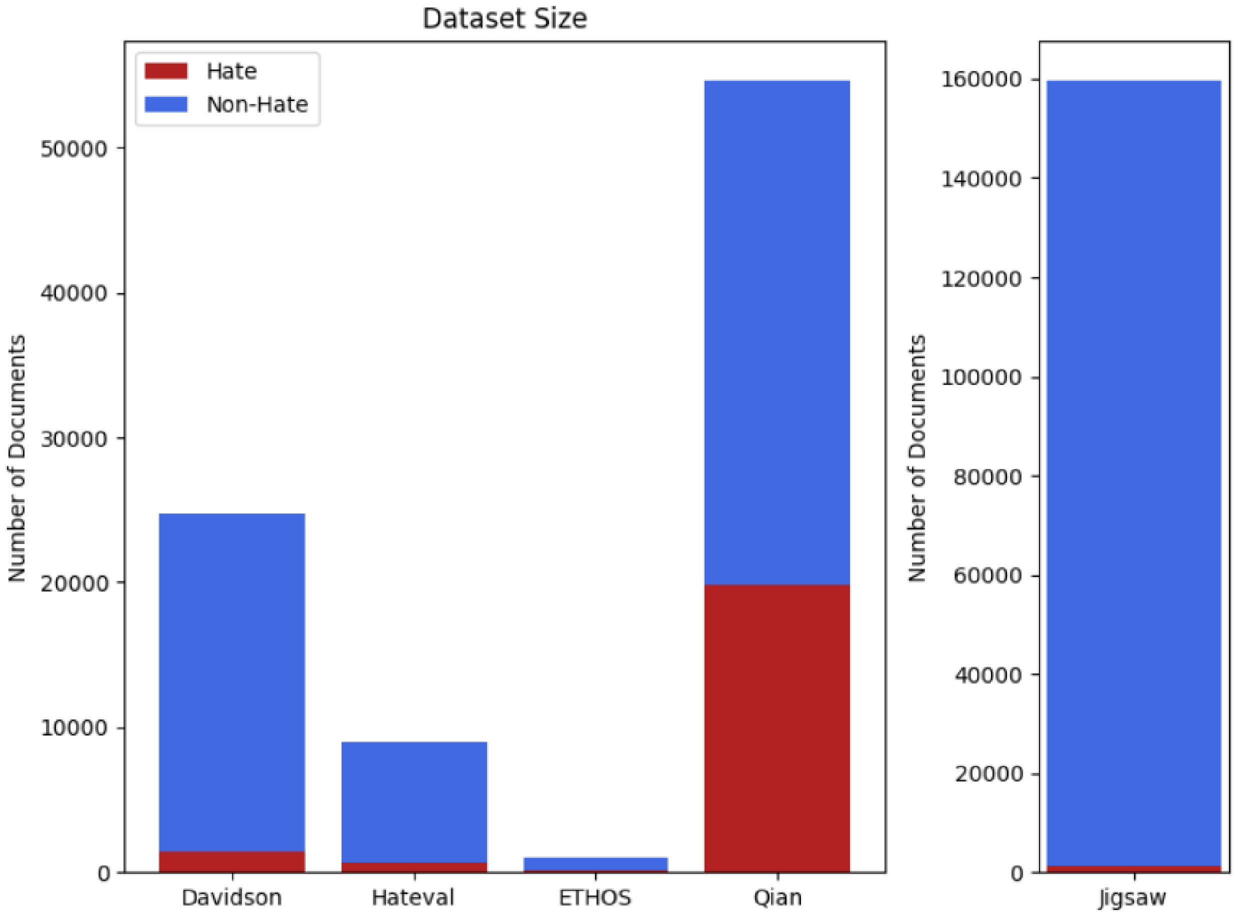

Figure 2: Dataset non-hate vs hate distribution

From the analysis presented in Figure 2, we can see that most datasets are composed mostly of non-hate documents. It is likely that further dataset augmentation will be necessary when training a hate speech classifier or else the performance on the hate class will be substantially subpar.

Average Document Length

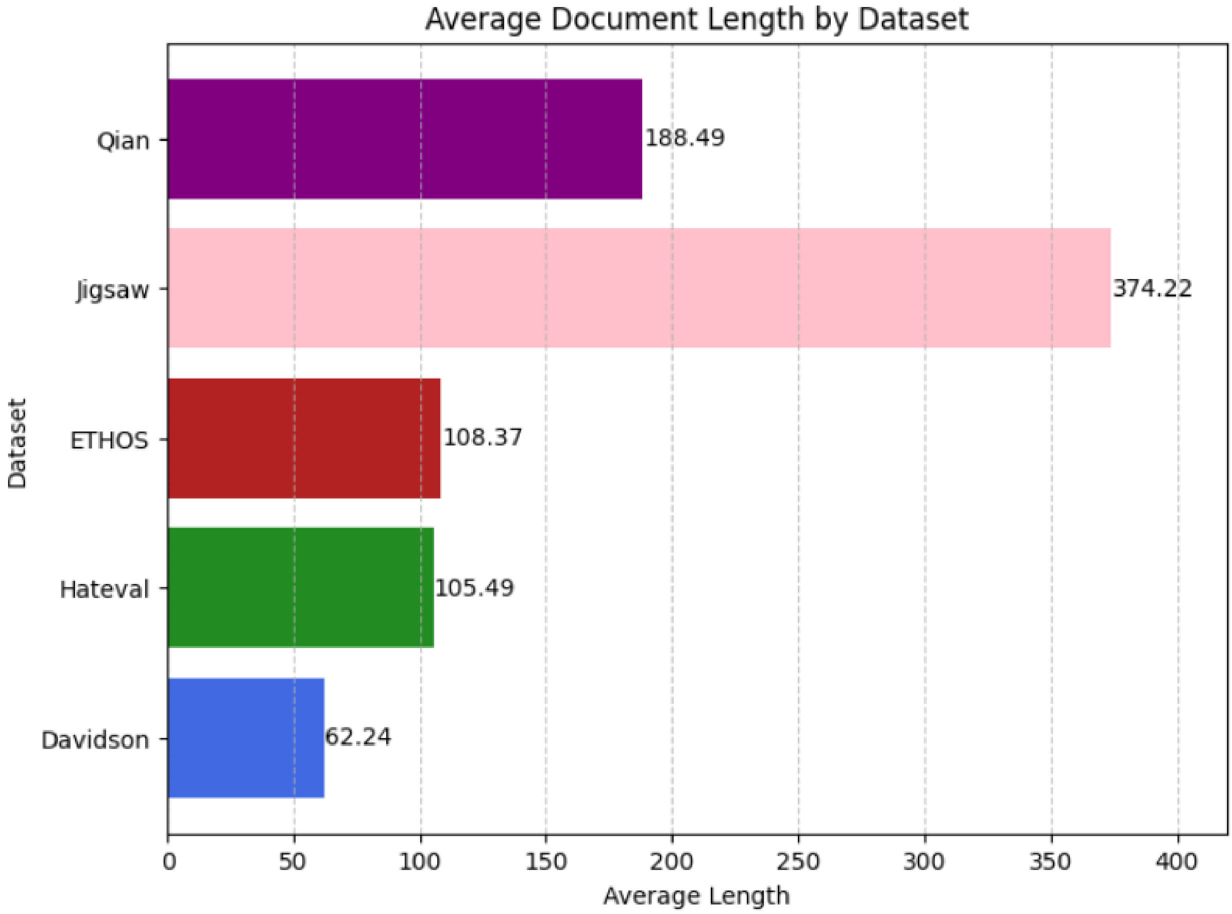

Figure 3: Average length of the documents in each dataset

Longer document length would provide better contextual data and likely increase model performance. This is especially the case in the context of hate speech detection, as hate speech is often intentionally subliminal or sarcastic, and can be difficult to identify with a lack of context. From our exploratory data analysis (Figures 2 & 3), the Qian dataset is likely to be the best-performing dataset, as it combines both long document length as well as a significantly higher number of hate class documents compared to the other datasets.

Dataset Content

A major issue in the hate speech detection field is the cross-dataset performance (Fortuna et al., 2021), which stems from the different ways the datasets are created, producing inherent bias. It will be important for the datasets in this project to cover a wide spectrum of hate speech, which will allow for the verification of cross-dataset performance.

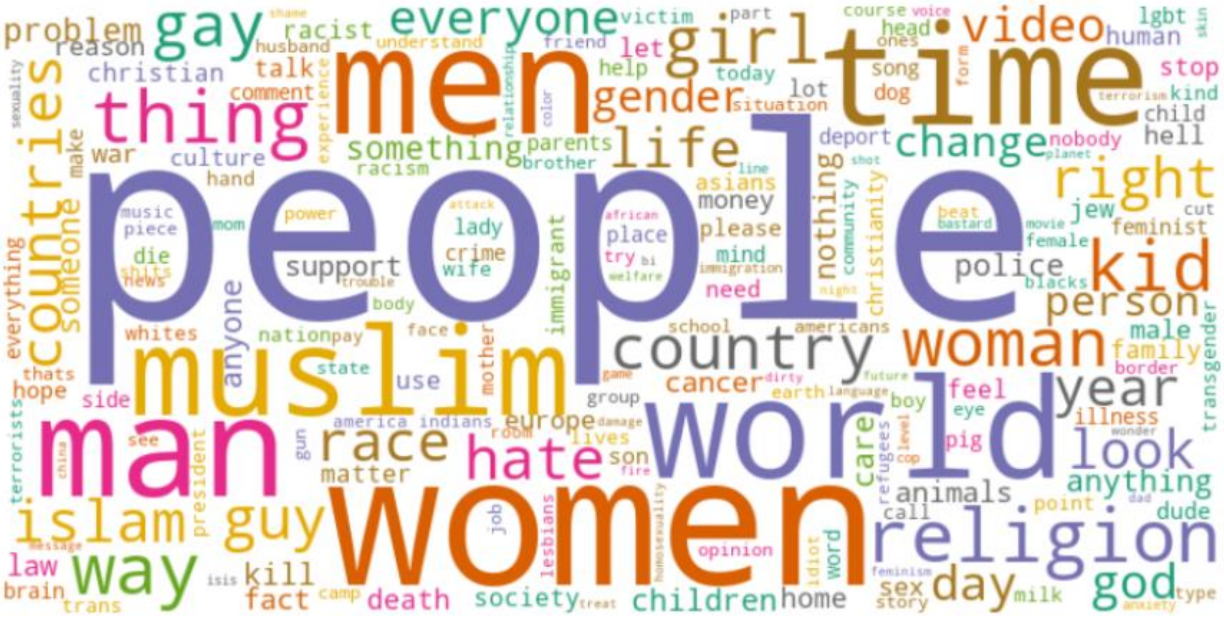

Figure 4: Top 10 most frequent words in hate classified documents in all datasets

After plotting a word cloud (Figure 4) using all documents in the ETHOS dataset, we can easily see the main targets and overall focus of the dataset. It is largely directed towards women as well as religious identity, with a focus on Islam.

To more efficiently understand the focus of each dataset, we will plot the 10 most frequent words found in the documents of each dataset that are classified as hate.

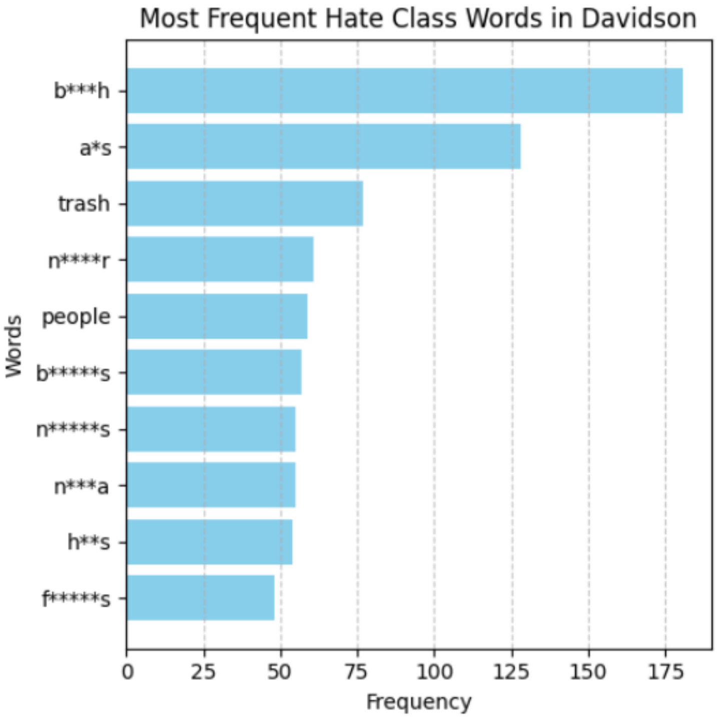

Figure 5: Top 10 most frequent words in hate classified documents in Davidson

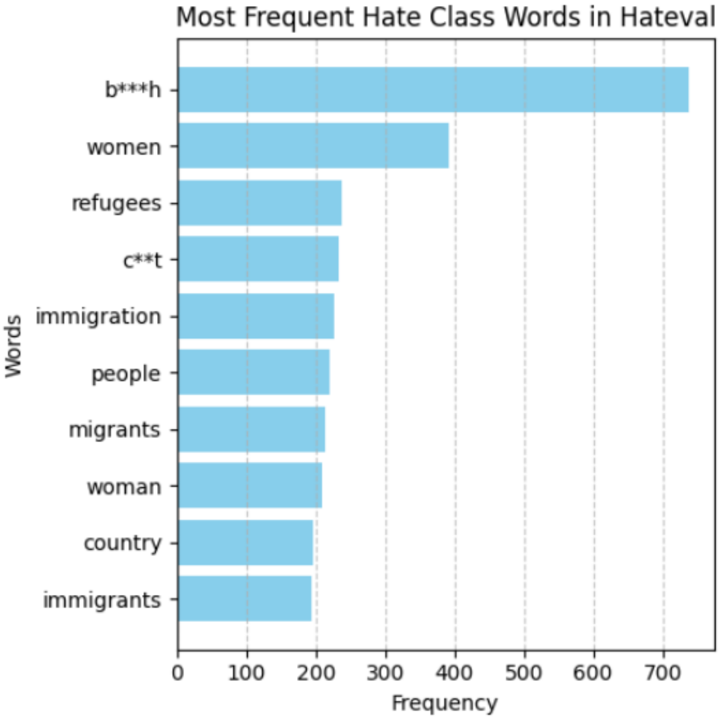

Figure 6: Top 10 most frequent words in hate classified documents in Hateval

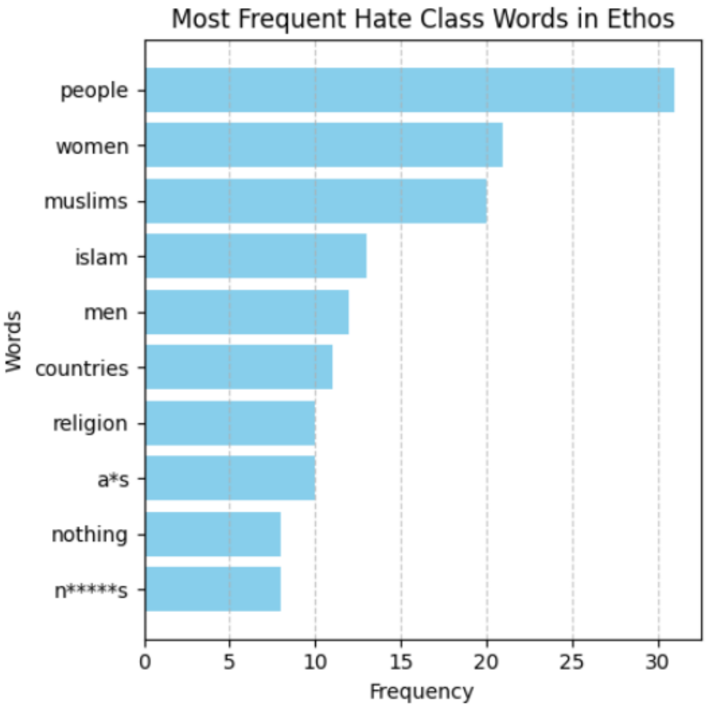

Figure 7: Top 10 most frequent words in hate classified documents in Ethos

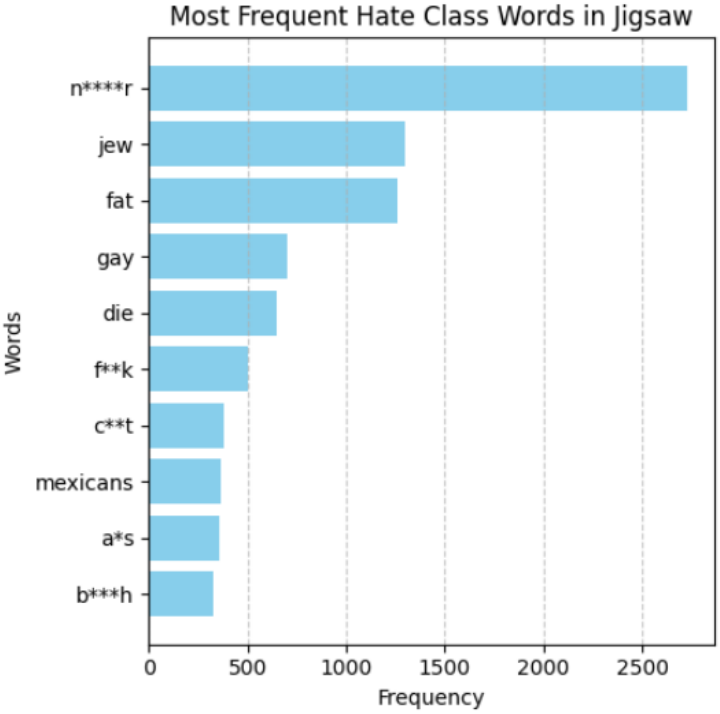

Figure 8: Top 10 most frequent words in hate classified documents in Jigsaw

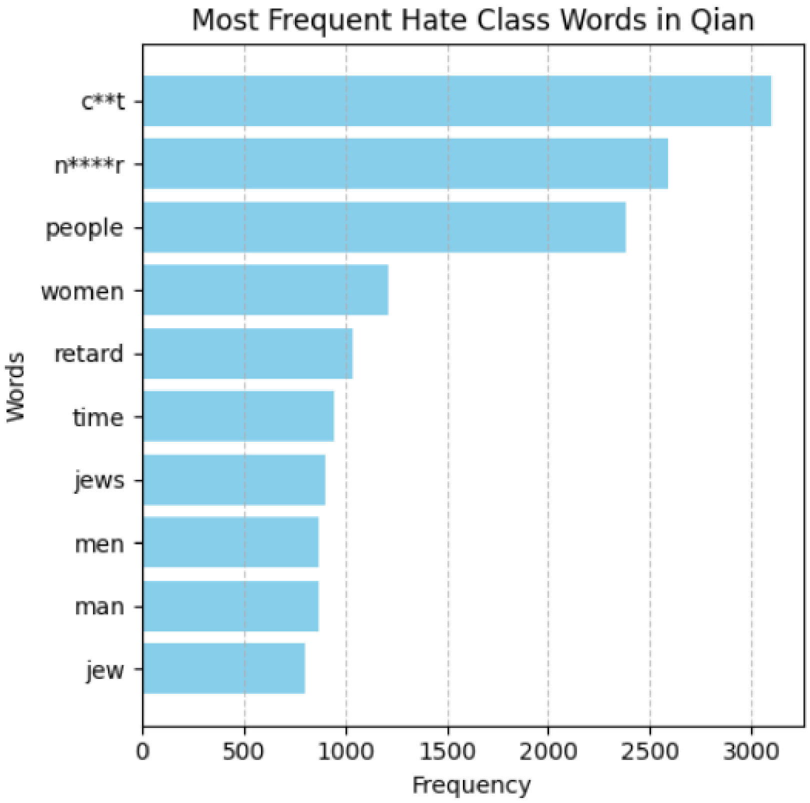

Figure 9: Top 10 most frequent words in hate classified documents in Qian

Most Frequent Words in Each Dataset

By examining the frequency plots (Figures 5-9), we can identify the potential focus of the hate speech of each dataset, as summarized below:

- Davidson: Race (mostly towards Black individuals), Homosexuals
- Hateval: Immigrants, Women
- ETHOS: Religion (mostly towards Islam), Women, Race
- Jigsaw: Race, Anti Semitism
- Qian: Race, Intellectual and Developmental disabilities, Women

The analysis shows varying focus in the content of each dataset. We can conclude that the challenge of developing a generalizable hate speech detector is likely to be significant due to the inherent bias of the datasets, and additional efforts in model augmentation and cross-dataset validation will be necessary to overcome this barrier.

Methodology and Experiments

Aim of Study

As stated earlier, the goal of this MRP is to analyze the performance of the state-of-the-art hate speech models from the literature for the purpose of improving their performance. A key component of this involves cross dataset testing to observe the influence of dataset bias. We will then attempt to alleviate the impact of bias through ensemble methods and dataset augmentation.

Classical Baseline

In earlier literature, classical machine learning methods were employed with success in hate speech classification Yin & Zubiaga (2021), Jahan & Oussalah (2023). They have since been outperformed by deep learning methods but are far faster to train. To provide a baseline of performance, several classical feature representation and classifier models were evaluated on the Davidson, Hateval and ETHOS datasets.

Classical Feature Representation

The feature representation methods evaluated were Term Frequency Inverse Document Frequency (TFIDF), Bag of Words (BOW) as well as Word2Vec (W2V). While Word2Vec uses a deep learning implementation, all 3 feature representation models were frequently seen being used in combination with classical classifier models according to Yin & Zubiaga, (2021) and Jahan & Oussalah, (2023). For this experiment, the TFIDF and BOW models were first fitted onto document strings and then used to transform the strings into feature vectors. For Word2Vec, the pretrained weights were used to transform the string into an array of word vectors; the word vectors were then averaged across each dimension to produce a single feature vector that summarizes the document. To evaluate the performance of the feature representation models, the experiment will only look at the results across each dataset classified with the same classifier model, this allows the feature representation results to be evaluated independently from the classifier models.

Classical Classifier Evaluation

The classical classifiers chosen were Logistic Regression (LR), Naïve Bayes (NB) and Support Vector Machines (SVM). To evaluate the classifiers, the feature vector data was first split into training and validation sets. 5-fold cross validation was then applied to the training set which was done to evaluate the weighted F1 score of every model combination on each dataset. The models were then trained on the full training set and evaluated on the validation set to determine their performance based on precision, recall and macro F1.

Deep Learning Model

With a baseline model selected, the next experiment involved the evaluation of the best performing deep learning model from the literature. This experiment evaluated the performance of deep learning feature representations models as well as downstream deep learning classifier architecture.

Deep Feature Representation

The deep learning feature representation models chosen were Word2Vec, BERT (Bidirectional Encoder Representations from Transformers), ALBERT (A Lite BERT) and ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately). To evaluate their performance, the downstream model was kept the same by using the Simple CNN model from Roy et al., (2020).

Deep Classifier Evaluation

To evaluate the performance of the downstream deep learning classifier models, a selection of models was chosen to verify the results from Zhang et al., (2018) and Roy et al. (2020). To ensure that the performance of the downstream model was isolated, all evaluations were conducted using Word2Vec as the upstream feature representation model. Listed below are the names used for the models in this project and their architecture:

Simple CNN

This is the best performing model from Roy et al. (2020), which is a 2-layer CNN with 100 filters and a kernel size of 4 using ReLU activation. The CNN layers are followed by a global pooling layer, which is then followed by a dense softmax layer for the final output.

Simple LSTM

Roy et al. (2020) found CNN outperformed RNN. This model will verify their results. It consists of an LSTM layer with 64 output units and is set to output at each time timestep. The LSTM layer is followed by a global pooling layer, which is then followed by a softmax dense layer for output.

2-3-4 CNN

One of the models of note in Roy et al. (2020) was a 2-layer CNN, where each layer used a combination of kernel sizes of 2, 3 and 4. The idea is that these kernel sizes act as bi, tri and quadgram feature extractions. This model is a smaller variant of the model from Roy et al. (2020). It features a single convolution layer with 100 filters per kernel size of 2, 3 and 4 using ReLU activation with 300 filters. The output of each filter is then concatenated together in a concatenate layer, which is then followed by a global pooling into dense softmax layer for output.

CNN GRU

This is the best performing model from Zhang et al., (2018). It features a convolution layer with 100 filters and a kernel size of 4 using ReLU activation. The convolutional layer is then fed into a max pooling layer with a poolsize of 4. The max pool is then followed by a GRU layer with 100 output units and set to output at each timestep. The GRU layer is finally followed by a global max pooling layer, which feeds into a softmax dense layer for output.

2-3-4 CNN LSTM

This is a model inspired by the models evaluated in the literature. It uses the 2-3-4 kernel CNN architecture from 2-3-4 CNN followed by LSTM layers. The key architecture difference is the inclusion of an LSTM layer with 32 output units that follow each convolutional layer. This places a unique LSTM layer behind the convolutional layer of each kernel size. All 3 LSTM layers are set to output at each timestep, of which their outputs are concatenated together and passed to a global max pooling layer. The global max pooling layer is then followed by a dense softmax layer for output.

All downstream models were evaluated using 5-fold cross validation on the Davidson, Hateval and ETHOS datasets. The best performing model was selected for the next experiment.

Cross Dataset Evaluation

To evaluate the performance of hate speech classifiers in generalized use, the best performing model from the previous experiment will be trained on each of the 5 datasets (Davidson, Hateval, ETHOS, Jigsaw, Qian) and then tested on the 4 datasets that the model was not trained on. This will show how much model performance degrades when applied in a new context different from its training data, the severity of which will show the potential effect of dataset bias.

Ensemble and Dataset Augmentation

The results from the previous experiment showed that model performance significantly degraded when applied across different datasets. To improve model performance, 3 new models were proposed and evaluated to alleviate the impact of dataset bias:

Dataset Augmentation

The most sensible first step in alleviating dataset bias is to increase the size of the dataset and include data that the dataset was biased against. This model makes no changes to the standalone classifier model from the cross-dataset evaluation but instead trains it on more datasets to reduce the bias in its training data.

Stacked Ensemble

This model fits a feed forward neural network with a hidden layer size of 10 on top of the individual predictions of the standalone classifier models trained in the cross-dataset evaluation. This neural network is then trained on all 5 datasets to classify hate speech based on the predictions of the standalone classifiers. This method is referred to as stacked ensemble, and has been shown to increase model performance in other NLP tasks (Akhtar et al., 2020).

Dynamic Ensemble

This novel ensemble model combines the output of its standalone classifiers differently compared to the stacked ensemble as follows. The standalone classifier predictions are combined into a single prediction by performing a weighted sum of the individual predictions. To determine the weights for each standalone classifier, however, we diverge from the static weights used in standard ensemble methods. A new model, combining TFIDF as the feature representation model with a feed forward neural network, consisting of hidden layers of size 5000, 1000 and 500 with ReLU activation layers in-between and a final softmax layer as output. This model predicts the weights used in the weighted sum in the ensemble model, which is calculated differently for each document. The intuition behind this model is the idea of a "manager" model, which can determine which standalone model should be trusted the most based on expertise and the information in the current document.

Proposed Model Experiments

The 3 proposed models were then used in '**All but One Experiment**'. This experiment trains each model on all datasets except one, and then evaluates its performance on the dataset exempted from the training. This will measure the performance of each model on unseen data and their effectiveness in reducing dataset bias

The second experiment conducted on the 3 proposed models is '**All Datasets Experiment**'. This experiment trains all 3 proposed models on all datasets and evaluates their performance on all datasets. Their performance will be compared to standalone models trained and applied to the same dataset. This experiment will measure how generalizable the proposed models are by comparing them with specialized models trained on a specific dataset.

Social Media Analysis

The final phase of this MRP project applies the individual classifiers in the cross-dataset evaluation and the stacked ensemble model to analyze a real-world dataset. During this phase, we classify a large sample of tweets in the 2021 #stopaapihate Twitter campaign and then use the predictions to determine the impact of hate speech by measuring its correlation with user engagement. To calculate user engagement with tweets, we sum the total likes, retweets, comments and quotes of each tweet into a single metric called engagement. This engagement metric is then used to calculate a correlation score with hate speech likelihood, as well as the time series analysis of the impact of hate on the overall campaign.

Results

Classical Baseline

Classical Feature Representation

The results from the classical feature representation experiment show Word2Vec consistently performing worse compared to TFIDF and BOW (Figure 10), which perform similarly. Looking at Appendix A - Table 1, a similar pattern can be observed in the Logistic Regression results.

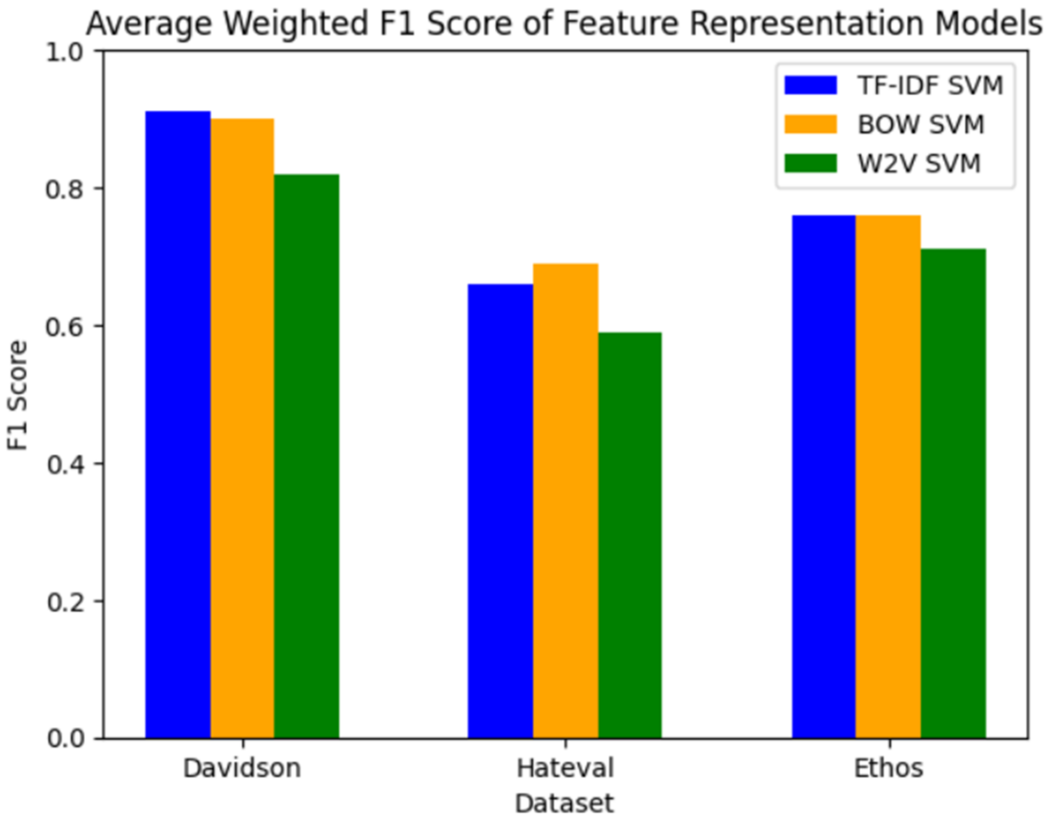

Figure 10: Cross validated average weighted F1 score of feature representation models.

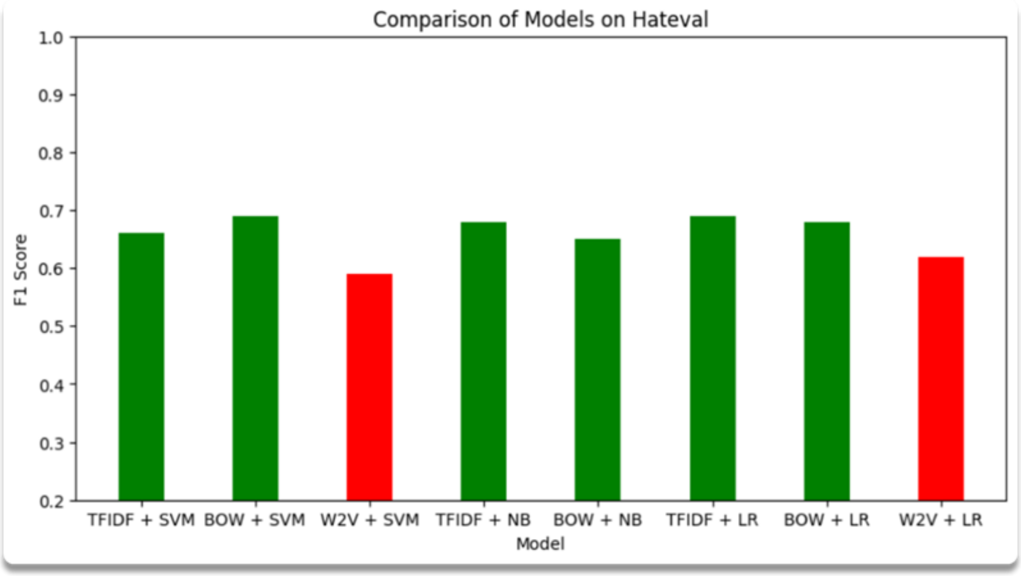

Figure 11: Comparison of cross validated averaged weighted F1 score of all models on the Hateval dataset

Comparing all models and isolating the Hateval dataset (Figure 11), the poor performance of Word2Vec is once again highlighted.

We can attribute this result to the averaging done on the word vectors computed by Word2Vec. This process effectively removes feature information from the data.

Without the ability to process the individual word vectors sequentially, classical models cannot leverage the contextual information encoded in individual word vectors. Taking the average of the word vectors then results in ineffective feature discrimination for their classification task.

Word2Vec will instead be tested again in the Deep Learning model evaluations, while TFIDF will be the selected classical baseline feature representation model. (BOW could also serve as the classical baseline, since its performance is nearly identical to TFIDF.)

Classical Classifier Evaluation

To analyze the classifier models, the classifiers are isolated by only using the bag of words feature representation model.

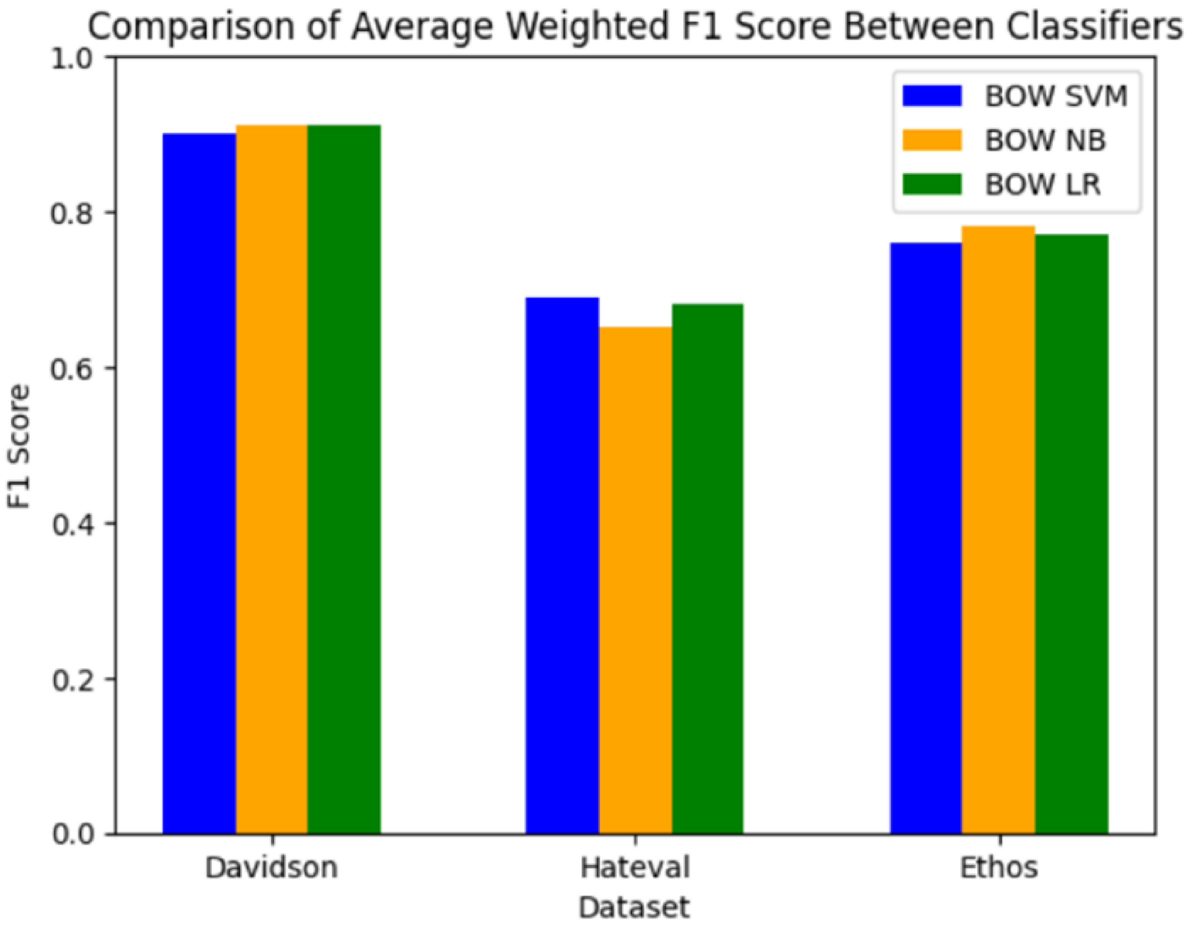

Figure 12: Comparison of cross validated average weighted F1 score between classifier models

Looking at the results (Figure 12), it is difficult to determine a better performing model, as all 3 models perform the same. It is important to note that the evaluation metric is weighted F1. Since non-hate classes make up the majority of the documents in all datasets, the weighted F1 performance will be dominated by the model's performance on the non-hate class. Since hate speech detectors are known to perform worse on the hate-class (Zhang & Luo, 2018), we will next compare the precision, recall and macro F1 of the 3 models on the hate class.

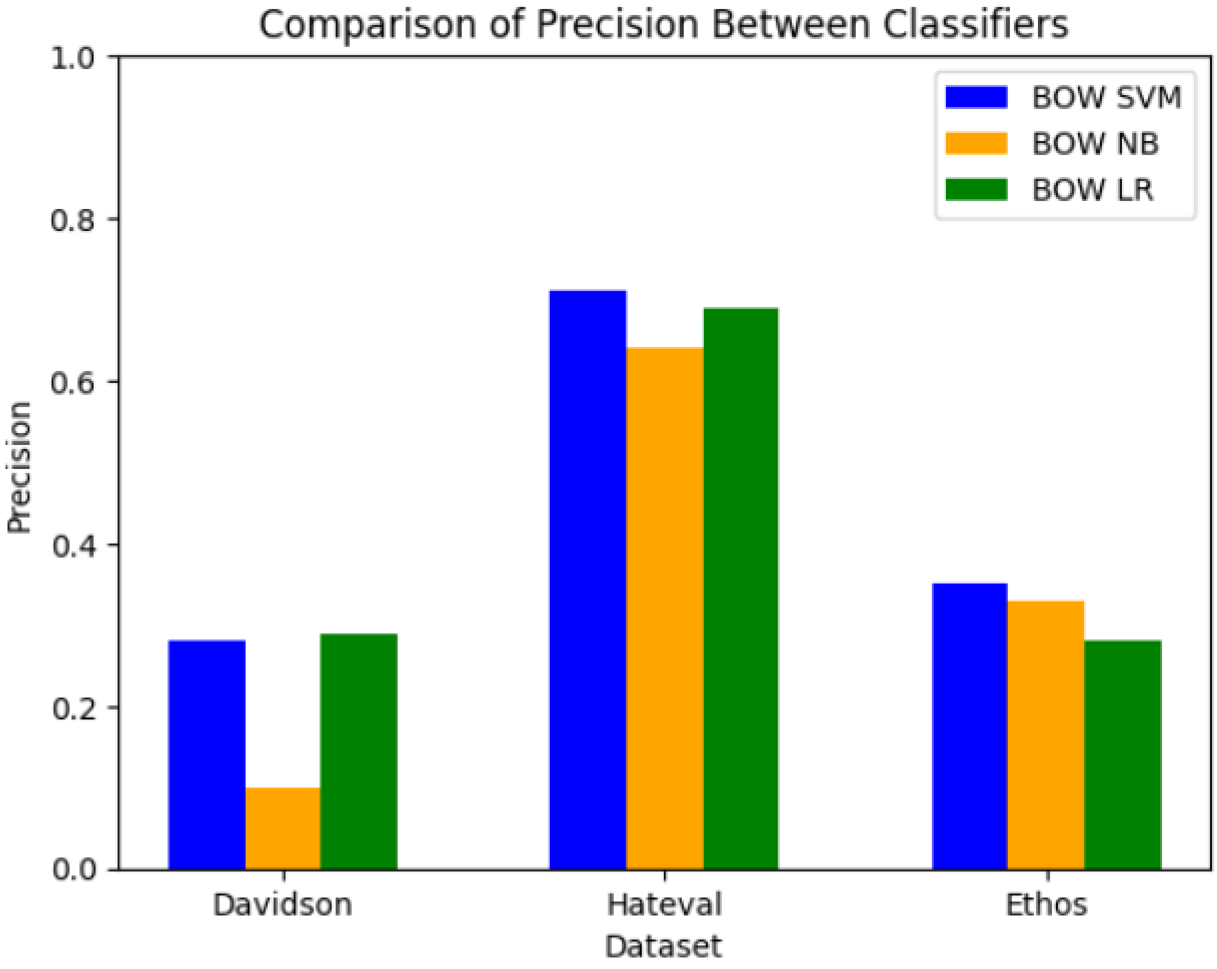

Figure 13: Comparison of precision performance between classifier models.

Analyzing precision performance on the hate class reveals a drastic change in the performance across models (Figure 13). The Naïve Bayes model performs significantly worse than Logistic Regression and SVM on the Davidson dataset. On Hateval and ETHOS, Naïve Bayes performs worse than SVM as well.

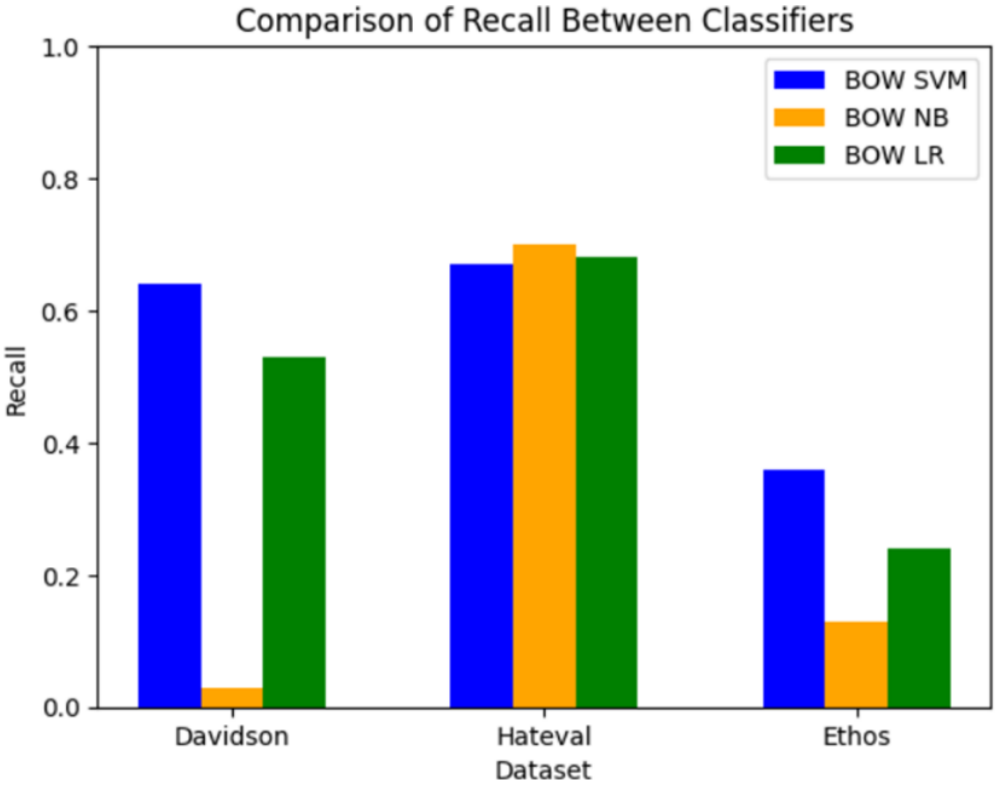

Figure 14: Comparison of recall performance between classifier models.

Analyzing recall performance shows the poor performance of the naive bayes model on 2 out of 3 datasets(Figure 14). This is likely attributed to the highly unbalanced classes in Davidson and ETHOS, resulting in the Naive Bayes classifier not being able to properly evaluate the contribution of each feature to the hate class.

SVM is once again the best performing model, performing better than logistic regression on the unbalanced datasets.

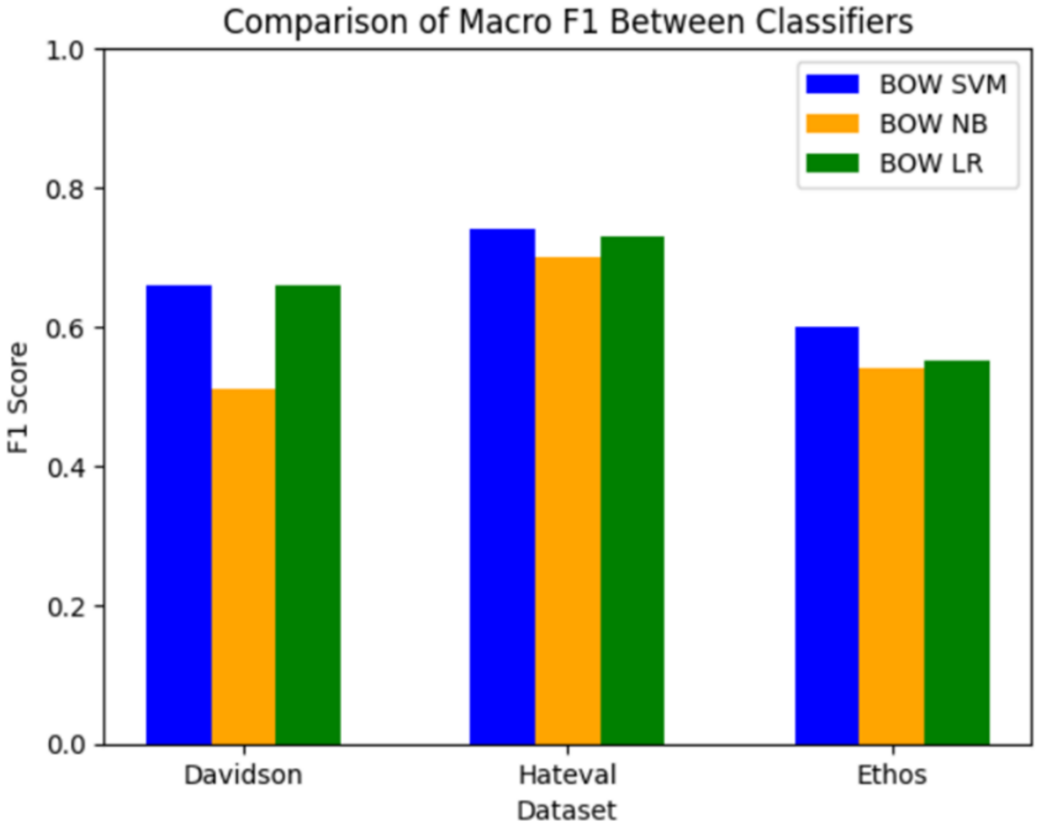

Figure 15: Comparison of macro F1 performance between classifier models.

Naive Bayes also underperforms based on macro F1. The results also suggest a slight advantage in performance for SVM over logistic regression.

In summary, the best performing classical model is shown to be TFIDF + SVM, which will be used as a baseline in the subsequent analysis with the deep learning models.

Deep Learning Results

Baseline Comparison

All deep learning models were evaluated using Word2Vec as their embedding model. Figure 16 shows that all deep learning models either perform similarly or outperform the TFIDF + SVM baseline in weighted F1. This assures that the following experiments are valid, and reaffirms the claim of the superior performance of deep learning models from (Yin & Zubiaga, 2021) and (Jahan & Oussalah, 2023).

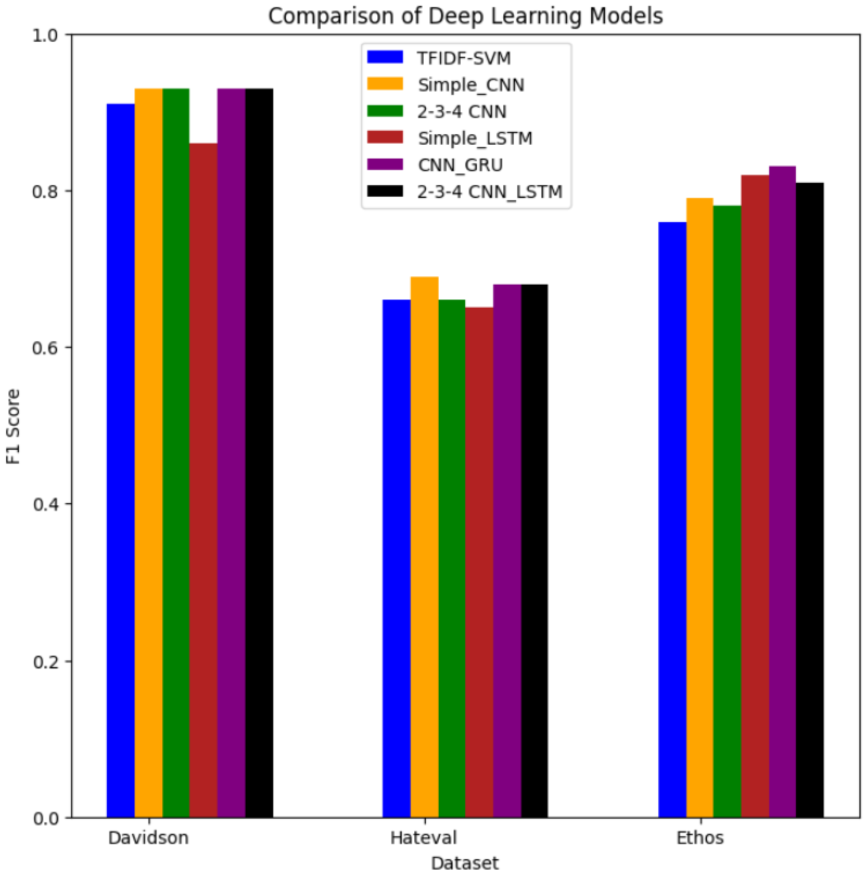

Figure 16: Weighted F1 Performance of Deep Learning Models and Classical Baseline

Embedding Evaluation

The overall best performing model of the embedding evaluation was ALBERT. In weighted F1 (Figure 17, Appendix A-4, A-8), ALBERT outperformed both BERT as well as ELECTRA across all datasets except for Davidson, where it performed similarly to BERT. Word2Vec also performed surprisingly well compared to the transformer models, even outperforming BERT on Hateval. However, when evaluating solely on the hate class precision (Appendix B-1, B-5), BERT is shown to have the overall best performance across the transformer models, with ALBERT consistently performing poorly. Focusing on recall precision (Figure 18, Appendix B-2, B-6) next, ALBERT and BERT are both performing well, with ALBERT significantly outperforming the other models on Hateval.

From these results, BERT is shown to perform the overall best when looking at performance in the hate class, while ALBERT performed the best when looking at overall performance across both classes.

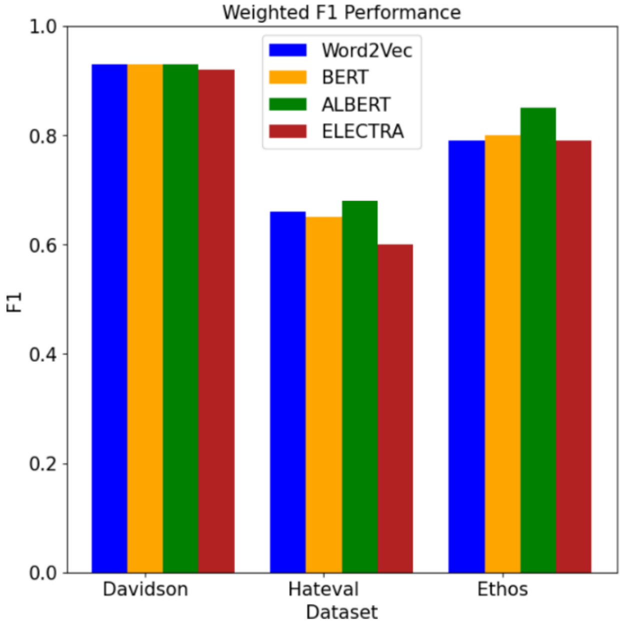

Figure 17: Weighted F1 Performance of Embedding Models

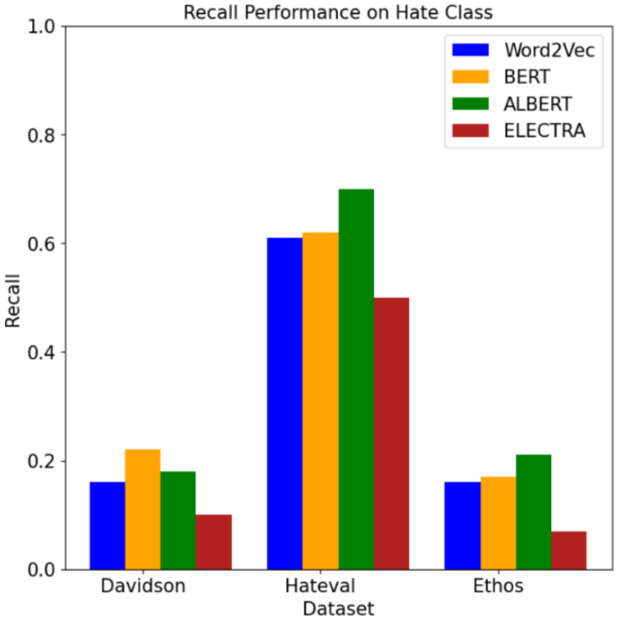

Figure 18: Recall Performance on Hate Class of Embedding Models

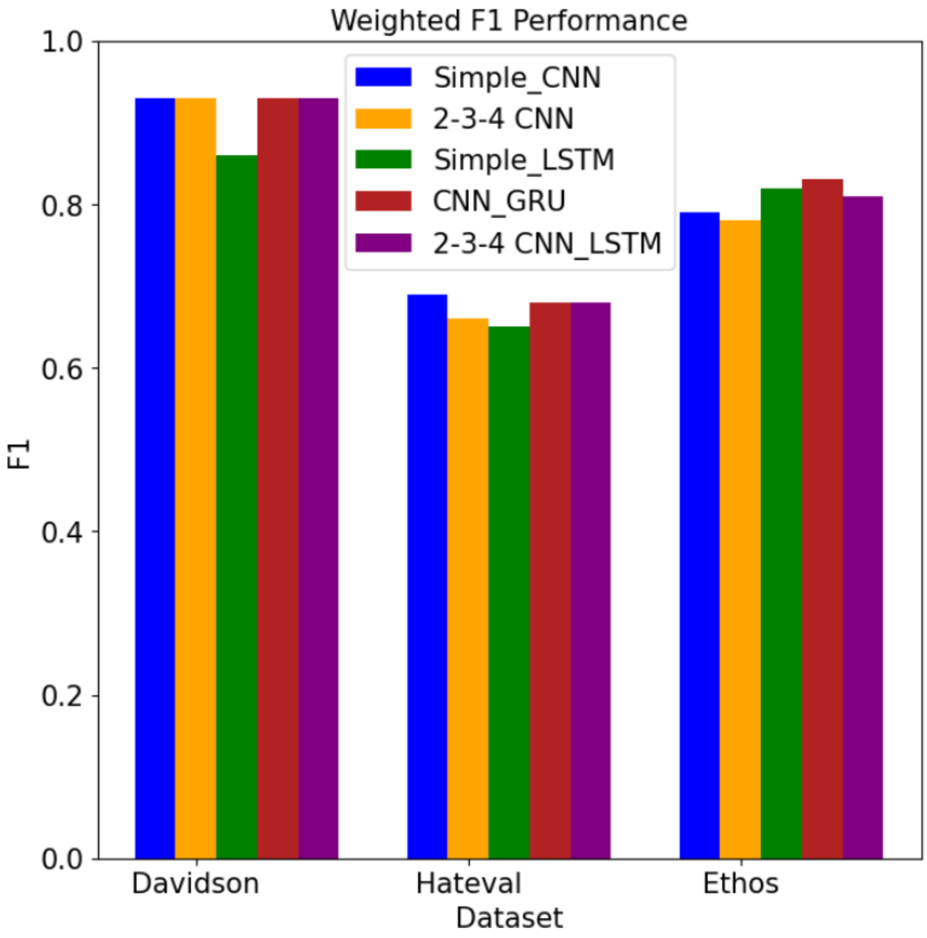

Figure 19: Weighted F1 Performance of Downstream Models

Downstream Evaluation

An initial analysis of weighted F1 performance (Figure 19, Appendix C-4, C-8) shows similar performance between all downstream models. The most significant observation is the lower performance of Simple LSTM, which is made especially clear by its performance on Davidson. Furthermore, the proposed model of this project, 2-3-4 CNN LSTM, performed similarly compared to models from the literature in this metric.

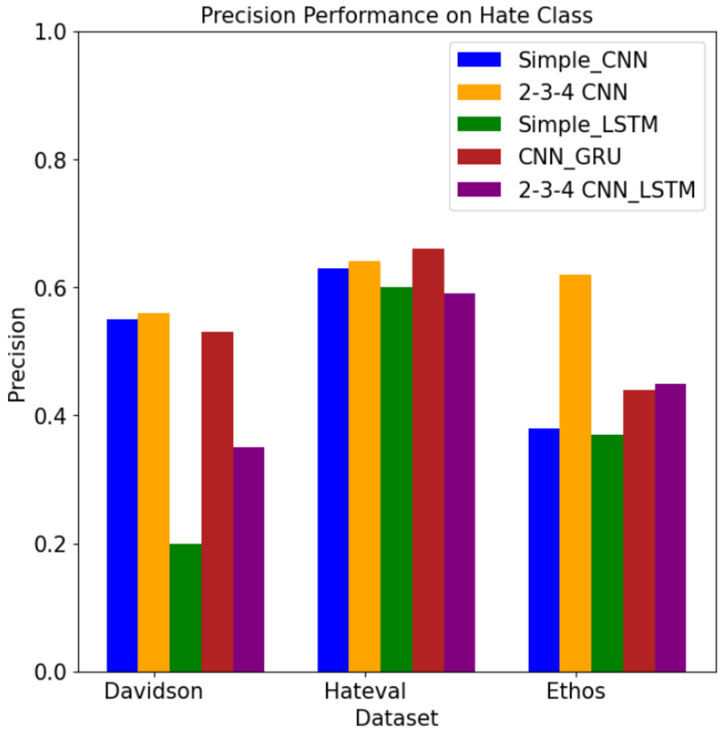

Figure 20: Precision Performance on Hate Class of Downstream Models

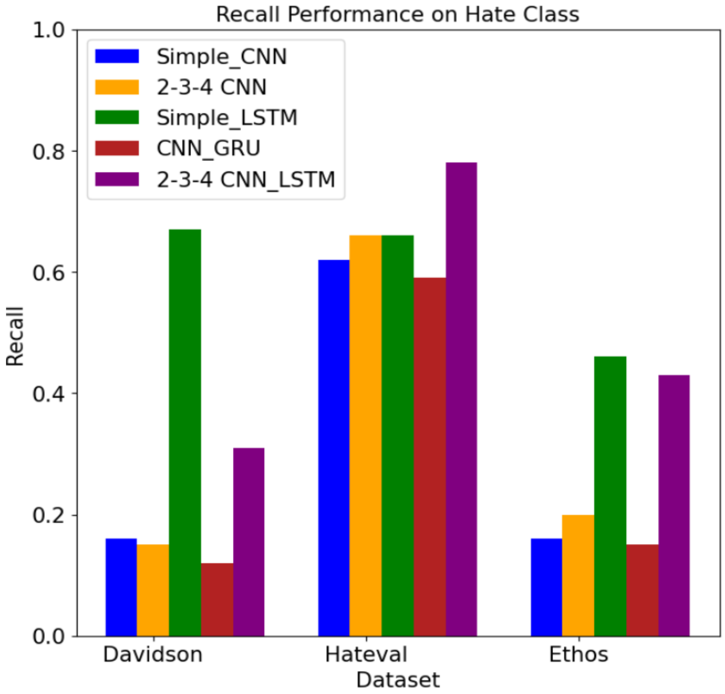

Figure 21: Recall Performance on Hate Class of Downstream Models

Further analysis of each model on the hate class reveals significant differences. In precision, the convolutional models outperformed the recurrent models. On ETHOS, 2-3-4 CNN outperformed every other model significantly and consistently outperformed Simple CNN across all datasets. CNN GRU (Zhang et al., 2018) from the literature outperformed Simple CNN (Roy et al., 2020) overall.

Looking at the recall performance (Figure 21, Appendix C-2,C-6), Simple LSTM and 2-3-4 CNN LSTM outperformed other convolutional models. However, this level of performance was not observed for CNN GRU; in fact, the GRU layer appears to reduce the recall of the model, resulting in Simple CNN outperforming CNN GRU. This discrepancy is likely associated with the architectural differences between LSTM and GRU, as LSTM models tend to fit better on smaller amounts of data due to their higher number of weights.

Deep Learning Conclusion

From the downstream model results, conclusions can be made about CNN, LSTM and GRU architecture in hate speech detection tasks. The different kernel sizes in 2-3-4 CNN appear to provide a slight advantage in precision compared to single kernel size models. LSTM layers provide better recall performance over convolutional layers while sacrificing precision performance. GRU layers may further enhance the precision performance of CNN standalone models, while sacrificing recall performance.

The 2-3-4 CNN LSTM model serves as a good alternative to the models from the literature when looking to increase recall while retaining overall model performance. The CNN GRU model from (Zhang et al., 2018) stands as a counterpart to 2-3-4 CNN LSTM, offering greater precision while retaining overall model performance.

In hate speech detection, it is more important to maximize hate precision, as misidentifying non-hate as hate on social media can elicit negative responses from users. From the embedding model experiment, BERT was found to be the best performing model in the hate class, while ALBERT was the overall best performing model. Thus, the model combination BERT + CNN-GRU was chosen as the model architecture to be used in the subsequent cross-dataset, ensemble and dataset augmentation experiments.

Cross Dataset Evaluation

This experiment trains the BERT + CNN + GRU model from the previous experiment on one of the 5 datasets and evaluates it on the other 4. The experiment intends to reveal how well it performs on unseen data. The results (Table 1) show a severe degradation in performance overall. Bias is evidently significant within hate speech datasets and state-of-the-art models can overfit within the bias of their training set and perform poorly on unseen data.

In particular, when evaluated on the Hateval dataset, the performance of models not trained on Hateval is consistently poor regardless of the training dataset. From our exploratory data analysis, Hateval was the only dataset that did not feature race as one of its primary topics. The effect of dataset bias is shown here, where the models trained on datasets that included race as one of their primary topics fail to identify nonracial hate speech consistently. This result also shows that state-of-the-art hate speech models focus heavily on vocabulary and not semantics when classifying hate speech documents.

	Weighted F1 Performance in Cross Dataset Application					
	Trained on:	Davidson	Hateval	ETHOS	Jigsaw	Qian
Applied To:						
Davidson			0.53	0.91	0.51	0.68
Hateval		0.53		0.48	0.64	0.59
ETHOS		0.77	0.91		0.62	0.78
Jigsaw		0.89	0.69	0.86		0.91
Qian		0.68	0.69	0.44	0.75	

Table 1: Cross validated weighted F1 performance of BERT + CNN + GRU model evaluated on all datasets

Ensemble and Dataset Augmentation

This experiment follows the previous experiment by testing the 3 proposed methods for improving performance in generalized use cases.

All But One Experiment

	Weighted F1 Performance in Cross Dataset Application									
	Trained on:	Davidson	Hateval	ETHOS	Jigsaw	Qian		Stacked Ensemble	Augmented Dataset	Dynamic Ensemble
Applied To:										
Davidson			0.53	0.91	0.51	0.68		0.73	0.81	0.62
Hateval		0.53		0.48	0.64	0.59		0.46	0.51	0.51
ETHOS		0.77	0.91		0.62	0.78		0.89	0.91	0.89
Jigsaw		0.89	0.69	0.86		0.91		0.82	0.82	0.78
Qian		0.68	0.69	0.44	0.75			0.81	0.82	0.75

Table 2. Same data from Table 1 with the addition of the 3 proposed models trained on all datasets except for the evaluation dataset

In this experiment, the 3 proposed models outperform the standalone models in most cases. The augmented dataset model slightly outperforms the stacked and dynamic ensemble models (Table 2). All 3 models still perform poorly on the Hateval dataset, however.

Dynamic ensemble performs the worst of the 3 proposed models. This is likely due to the weight selection model producing inaccurate weights on top of the base ensemble models not featuring the evaluation data.

The results indicate that the proposed models offer increased performance on unseen data given that the unseen data is similar to the training data. The performance on Hateval shows that if the unseen data is significantly different, the proposed models offer no increase in performance.

All Datasets Experiment

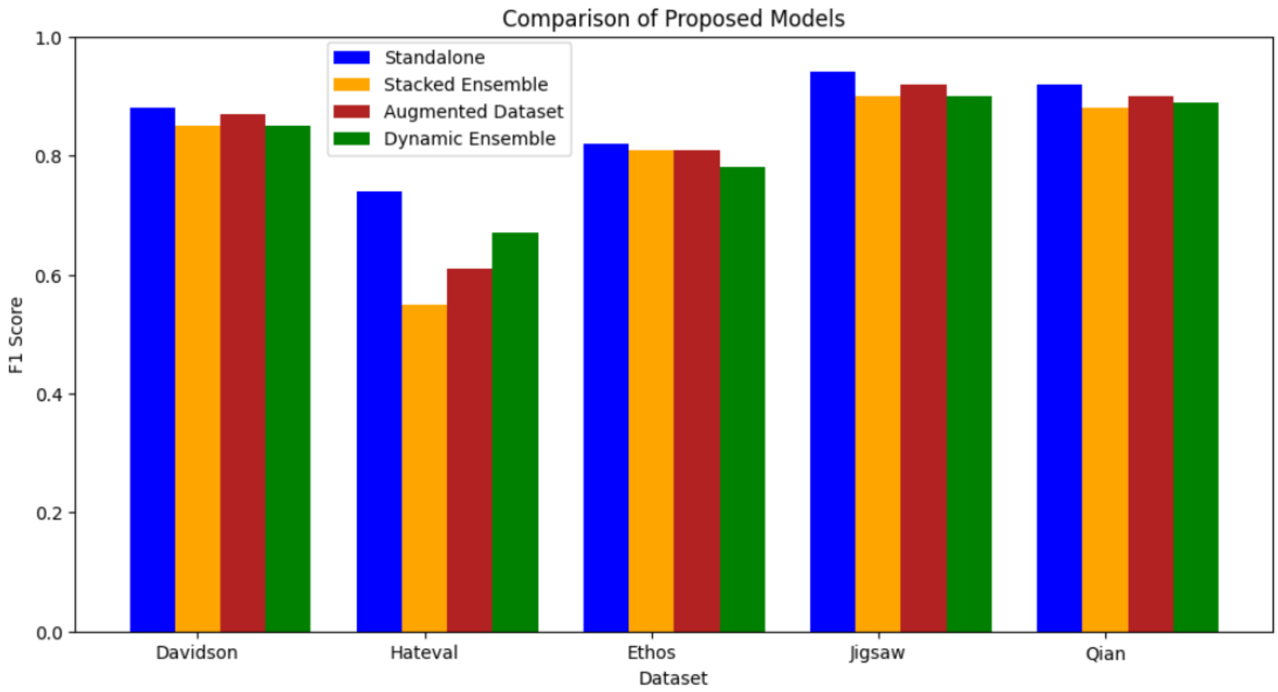

Figure 22: Comparison of standalone model trained on the same dataset with proposed models trained on all datasets

In this experiment, the stacked ensemble and augmented dataset models both perform reasonably close to the standalone models but show a severe decrease in performance on Hateval (Figure 22). Dynamic ensemble is seen significantly outperforming both stacked ensemble and the augmented dataset model on Hateval. The success of the dynamic ensemble model is due to the weight selection model deprioritizing the underlying classifiers that are not trained on Hateval. Thus, despite the total training data for the dynamic ensemble being heavily dominated by data not from the Hateval dataset, the dynamic ensemble model can still recover the performance of the standalone specialized model trained solely on Hateval. This result is significant as issues with dataset imbalance and bias can be alleviated with this novel dynamic ensemble method.

Social Media Analysis

Time Series Analysis

Continuing to use the ensemble predictions, we can determine the percentage of of tweets identified as hate speech on each day of the campaign (Figure 23). We can compare this trend with the total tweet count and engagement as seen in Figures 24-25. This allows us to construct the relation of hate and non-hate content in the social media campaign

From the tweet count and total engagement, we can see that the campaign kicks off on the 14th of March. The campaign momentum then slowly dies out over the rest of the month, with engagement decreasing faster than tweet count. We can also see that the total amount of hate speech is at its lowest at the beginning of the campaign and peaks several days after March 21st. The second spike in total tweet count also takes place a day after the spike in hate speech on the 22nd, which drives the total hate speech down once again. Afterwards, the campaign continues to decrease in momentum, and the total daily hate speech picks back up as well.

We can gather from this data that the start of the campaign is the most successful phase of the campaign because of the high number of campaign supporters initially. The following days of the campaign may invite an increasing amount of hate speech, which can then prompt campaign supporters to return to the campaign to resist the hate speech. Eventually, the volume of posts supporting the campaign decrease due to disinterest, while nefarious posts continue to circulate. The result is the closing phases of the campaign contain a higher percentage of hate compared to the beginning of the campaign.

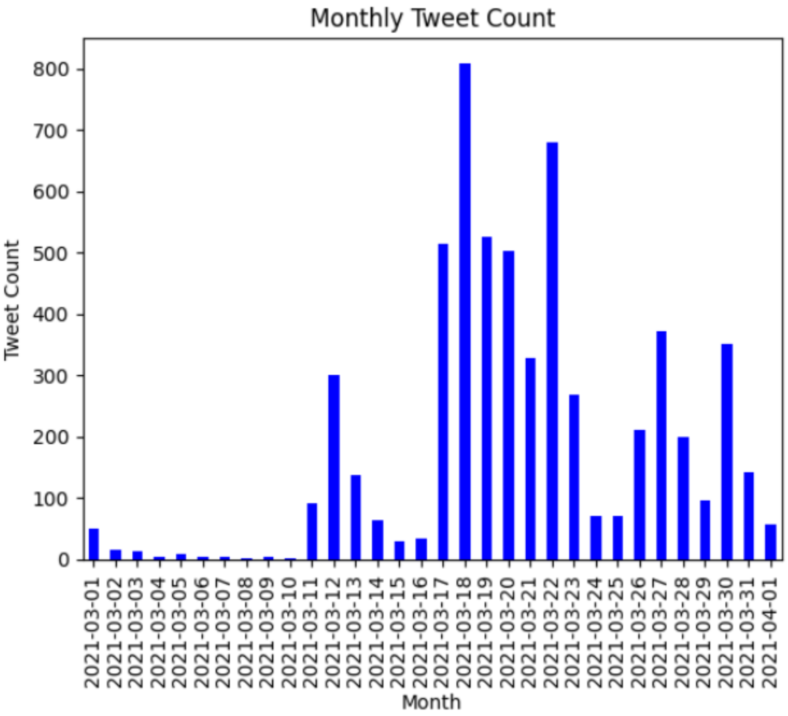

Figure 23: Monthly tweet count of the #stopaapihate social media campaign

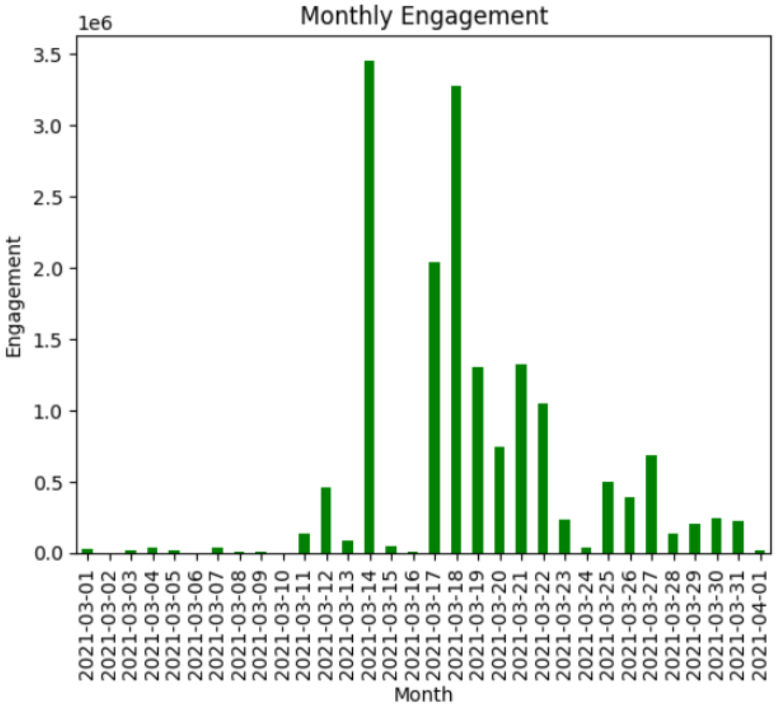

Figure 24: Monthly tweet engagement of the #stopaapihate social media campaign

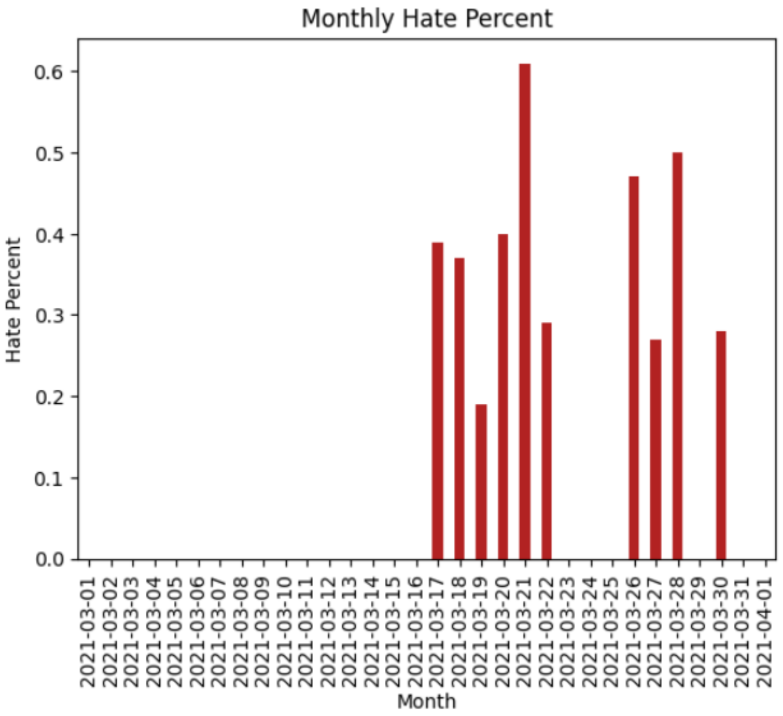

Figure 25: Monthly hate percent of the #stopaapihate social media campaign

Correlation

The correlation between the hate class and tweet engagement (Table 3) indicates a negligeable correlation between the 2 metrics. This suggests that for the vast majority of posts, non-hate and hate posts received similar amounts of engagement.

Correlation Between Hate Class and User Engagement						
Trained on:	Davidson	Hateval	ETHOS	Jigsaw	Qian	Ensemble
	0.000855	-0.001387	-0.001333	-0.002880	-0.002176	-0.003556

Table 3 Correlation between hate class and user engagement

Conclusion

This paper revealed many insights in the field of hate speech detection. We determined TFIDF / BOW + SVM to be the overall best performing classical hate speech detection model, while also showing that deep learning models consistently outperformed classical models. We also showed that deep learning architecture can have varying levels of impact on the performance metrics of the model. ALBERT was shown to be the best performing embedding model when assessing both hate and non-hate class performance, while BERT showed the best performance on the hate class. CNN layers were shown to improve model precision on the hate class, while RNN layers improved model recall. Specifically, for RNNs, LSTM was shown to overfit compared to GRU, leading to losses in precision and increased recall in the hate class. We thus concluded that BERT + CNN + GRU was the best model for maximizing precision performance on the hate class, as hate class precision was deemed the most valuable metric in hate speech detection.

We then evaluated the BERT + CNN + GRU model in generalized cases, which revealed degrading performance when evaluated outside of the training dataset. Our experiments showed that training datasets can be highly biased, and the resulting model can suffer in performance when exposed to different biases in other datasets. This result also shows that there are many subdomains of hate speech, and the curation of a single all-encompassing dataset capturing all subdomains is likely very difficult.

To alleviate the issue of dataset bias, we then proposed 3 models. Both stacked ensemble and dataset augmentation improved overall performance when evaluated on new datasets. Given that the new data is similar to the training data, both stacked ensemble and dataset augmentation offered increased performance. When the new data is significantly different from the training data, however, both stacked ensemble and dataset augmentation failed to achieve increased performance. Even after including the unseen data in the training data, both models still failed to recover the performance of a standalone model trained on the unseen data.

The novel model examined in this project, dynamic ensemble, was shown to successfully recover the performance of the standalone model on the unseen data. This model alleviates the issue of unbalanced domains and bias in the training data and can provide a generalized hate speech prediction that outperforms the state-of-the-art models trained on the same data. This result is significant, with the ensemble nature of the dynamic ensemble allowing for additional classifiers that feature new subdomains of hate speech to be added to the generalized model without sacrificing model performance in other hate speech subdomains. Dynamic ensemble thus serves a scalable and adaptable model for generalizable hate speech detection.

Appendix

A - Classical Baseline Results

Cross Validated Weighted F1 Performance			
Model / Dataset	Davidson	Hateval	ETHOS
TFIDF + SVM	**0.91**	**0.66**	**0.76**
BOW + SVM	**0.90**	**0.69**	**0.76**
W2V + SVM	**0.82**	**0.59**	**0.71**
TFIDF + NB	**0.91**	**0.68**	**0.76**
BOW + NB	**0.91**	**0.65**	**0.78**
TFIDF + LR	**0.91**	**0.69**	**0.76**
BOW + LR	**0.91**	**0.68**	**0.77**
W2V + LR	**0.83**	**0.62**	**0.72**

Table 4, Appendix A-1: Cross validated weighted F1 performance of each model combination on each dataset

Precision Performance on Hate Class			
Model / Dataset	Davidson	Hateval	ETHOS
TFIDF + SVM	**0.30**	**0.69**	**0.35**
BOW + SVM	**0.28**	**0.71**	**0.35**
W2V + SVM	**0.14**	**0.54**	**0.18**
TFIDF + NB	**0.60**	**0.74**	**0.00**
BOW + NB	**0.10**	**0.64**	**0.33**
TFIDF + LR	**0.28**	**0.69**	**0.39**
BOW + LR	**0.29**	**0.69**	**0.28**
W2V + LR	**0.16**	**0.62**	**0.28**

Table 5, Appendix A-2: Precision performance on the hate class of each model combination on each dataset.

Recall Performance on Hate Class			
Model / Dataset	Davidson	Hateval	ETHOS
TFIDF + SVM	**0.62**	**0.70**	**0.50**
BOW + SVM	**0.64**	**0.67**	**0.36**
W2V + SVM	**0.64**	**0.55**	**0.27**
TFIDF + NB	**0.01**	**0.57**	**0.00**
BOW + NB	**0.03**	**0.70**	**0.13**
TFIDF + LR	**0.56**	**0.70**	**0.40**
BOW + LR	**0.53**	**0.68**	**0.24**
W2V + LR	**0.64**	**0.62**	**0.79**

Table 6, Appendix A-3: Recall performance on the hate class of each model combination on each dataset.

Macro F1 Performance			
Model / Dataset	Davidson	Hateval	ETHOS
TFIDF + SVM	**0.68**	**0.73**	**0.63**
BOW + SVM	**0.66**	**0.74**	**0.60**
W2V + SVM	**0.55**	**0.60**	**0.52**
TFIDF + NB	**0.49**	**0.71**	**0.46**
BOW + NB	**0.51**	**0.70**	**0.54**
TFIDF + LR	**0.66**	**0.73**	**0.63**
BOW + LR	**0.66**	**0.73**	**0.55**
W2V + LR	**0.56**	**0.67**	**0.60**

Table 7, Appendix A-4: Macro F1 performance of each model combination on each dataset.

B - Embedding Evaluation

Precision on Hate Class			
Model	Dataset		
	Davidson	Hateval	Ethos
Word2Vec	0.55	0.64	0.38
BERT	0.55	0.65	0.33
ALBERT	0.53	0.61	0.20
ELECTRA	0.43	0.64	0.33

Table 8, Appendix B-1: Precision of Embedding Models on Hate Class

Recall on Hate Class			
Model	Dataset		
	Davidson	Hateval	Ethos
Word2Vec	0.16	0.61	0.16
BERT	0.22	0.62	0.17
ALBERT	0.18	0.70	0.21
ELECTRA	0.10	0.50	0.07

Table 9, Appendix B-2: Recall of Embedding Models on Hate Class

Macro F1 Performance			
Model	Dataset		
	Davidson	Hateval	Ethos
Word2Vec	0.60	0.63	0.56
BERT	0.64	0.62	0.56
ALBERT	0.62	0.65	0.56
ELECTRA	0.56	0.58	0.51

Table 10, Appendix B-3: Macro F1 Performance of Embedding Models

Weighted F1 Performance			
Model	Dataset		
	Davidson	Hateval	Ethos
Word2Vec	0.93	0.66	0.79
BERT	0.93	0.65	0.80
ALBERT	0.93	0.68	0.85
ELECTRA	0.92	0.60	0.79

Table 11, Appendix B-4: Weighted F1 Performance of Embedding Models

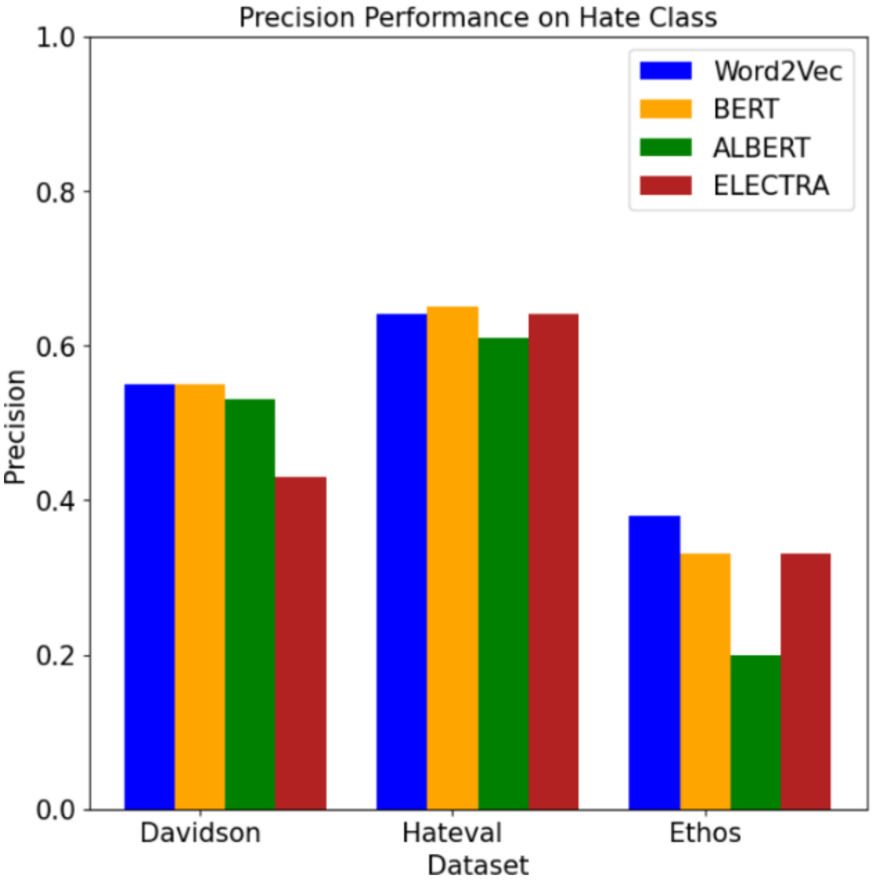

Figure 26, Appendix B-5: Precision Performance on Hate Class of Embedding Models

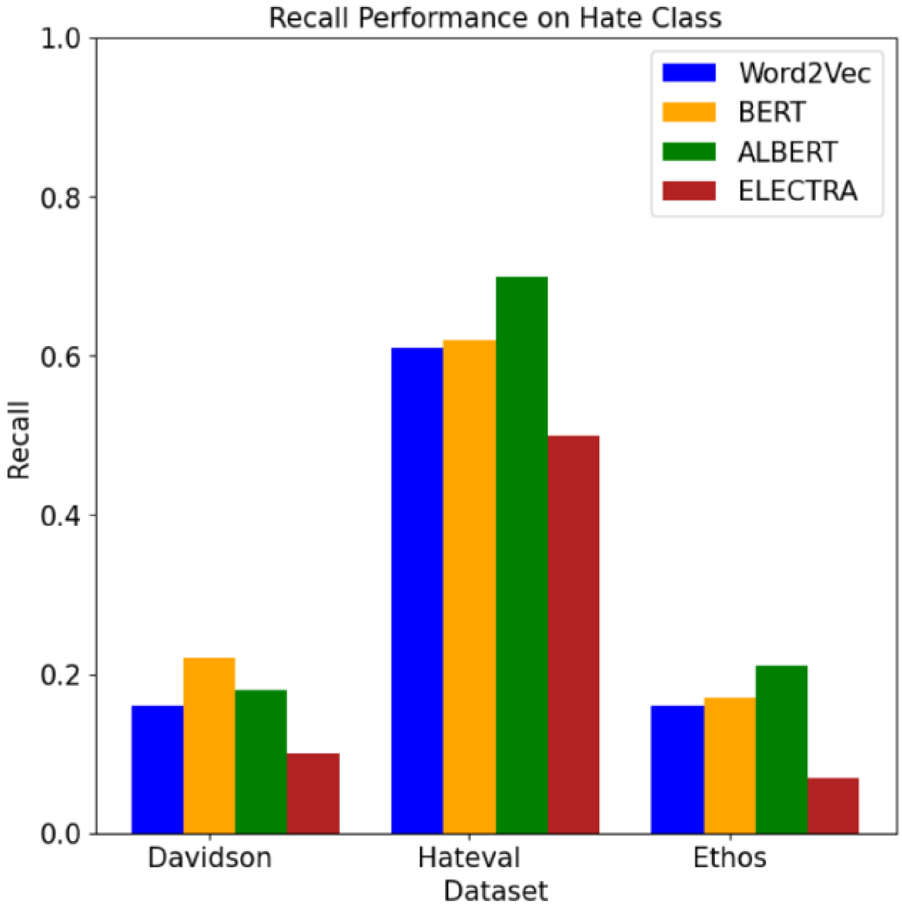

Figure 27, Appendix B-6: Recall Performance on Hate Class of Embedding Models

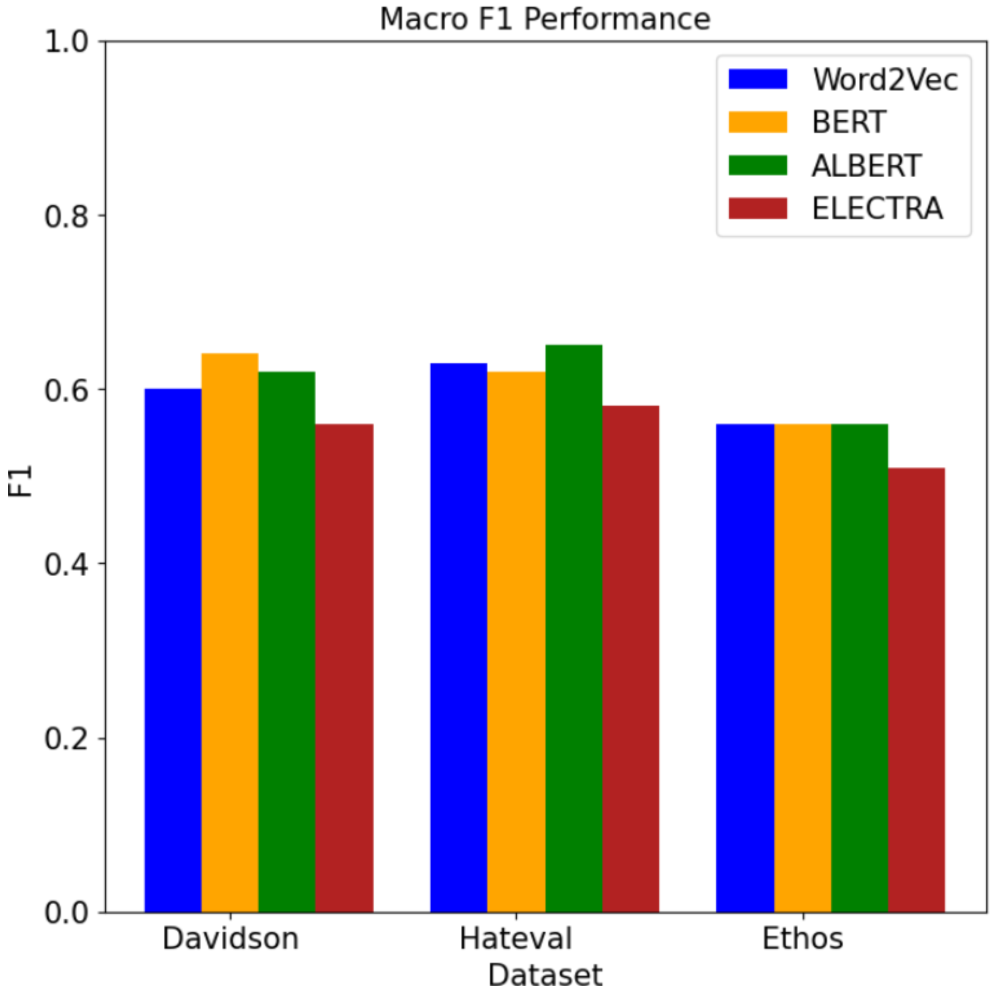

Figure 28, Appendix B-7: Macro F1 Performance of Embedding Models

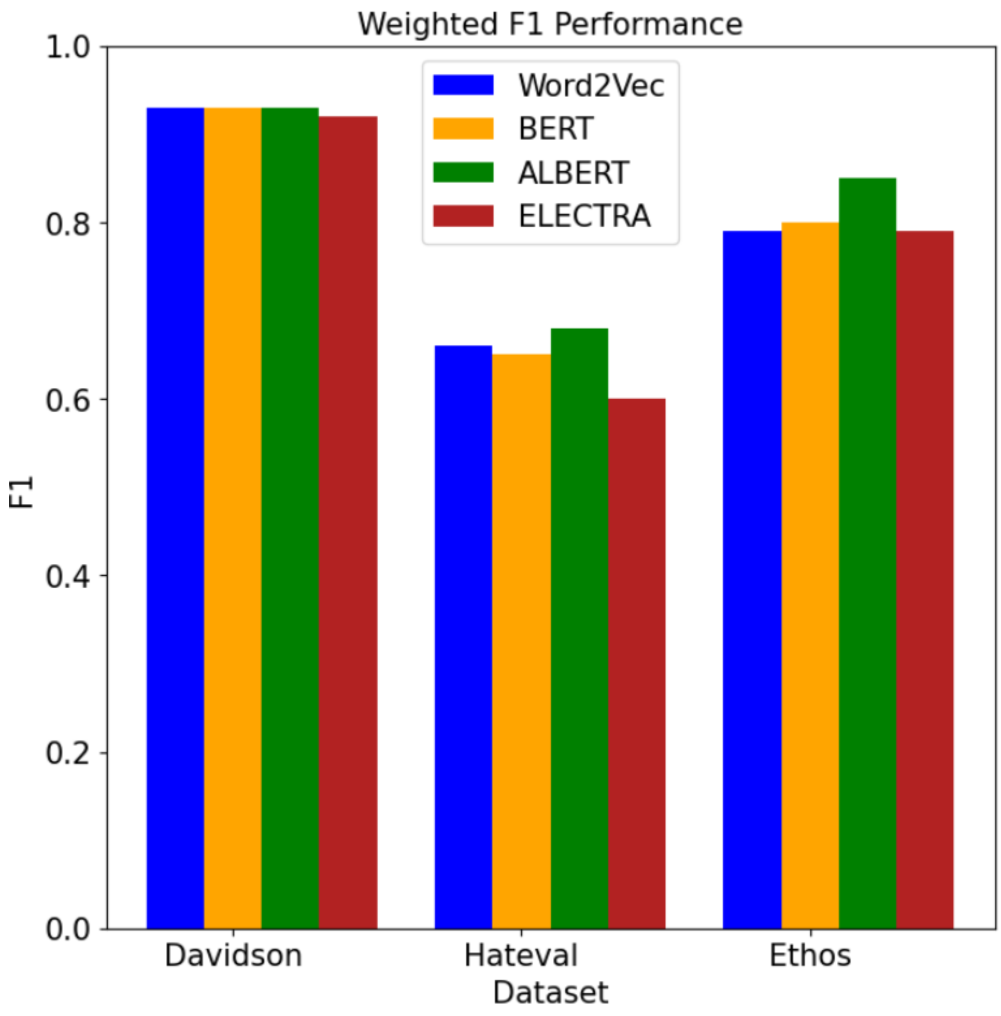

Figure 29, Appendix B-8: Weighted F1 Performance of Embedding Models

C - Downstream Evaluation

Precision on Hate Class			
Model	Dataset		
	Davidson	Hateval	Ethos
Simple CNN	0.55	0.63	0.38
2-3-4 CNN	0.56	0.64	0.62
Simple LSTM	0.20	0.60	0.37
CNN GRU	0.53	0.66	0.44
2-3-4 CNN LSTM	0.35	0.59	0.45

Table 12, Appendix C-1: Precision of Downstream Models on Hate Class

Recall on Hate Class			
Model	Dataset		
	Davidson	Hateval	Ethos
Simple CNN	0.16	0.62	0.16
2-3-4 CNN	0.15	0.66	0.20
Simple LSTM	0.67	0.66	0.46
CNN GRU	0.12	0.59	0.15
2-3-4 CNN LSTM	0.31	0.78	0.43

Table 13, Appendix C-2: Recall of Downstream Models on Hate Class

Macro F1 Performance			
Model	Dataset		
	Davidson	Hateval	Ethos
Simple CNN	0.60	0.65	0.56
2-3-4 CNN	0.60	0.64	0.60
Simple LSTM	0.60	0.61	0.65
CNN GRU	0.58	0.64	0.57
2-3-4 CNN LSTM	0.64	0.65	0.66

Table 14, Appendix C-3: Macro F1 Performance of Downstream Models

Weighted F1 Performance			
Model	Dataset		
	Davidson	Hateval	Ethos
Simple CNN	0.93	0.69	0.79
2-3-4 CNN	0.93	0.66	0.78
Simple LSTM	0.86	0.65	0.82
CNN GRU	0.93	0.68	0.83
2-3-4 CNN LSTM	0.93	0.68	0.81

Table 15, Appendix C-4: Weighted F1 Performance of Downstream Models

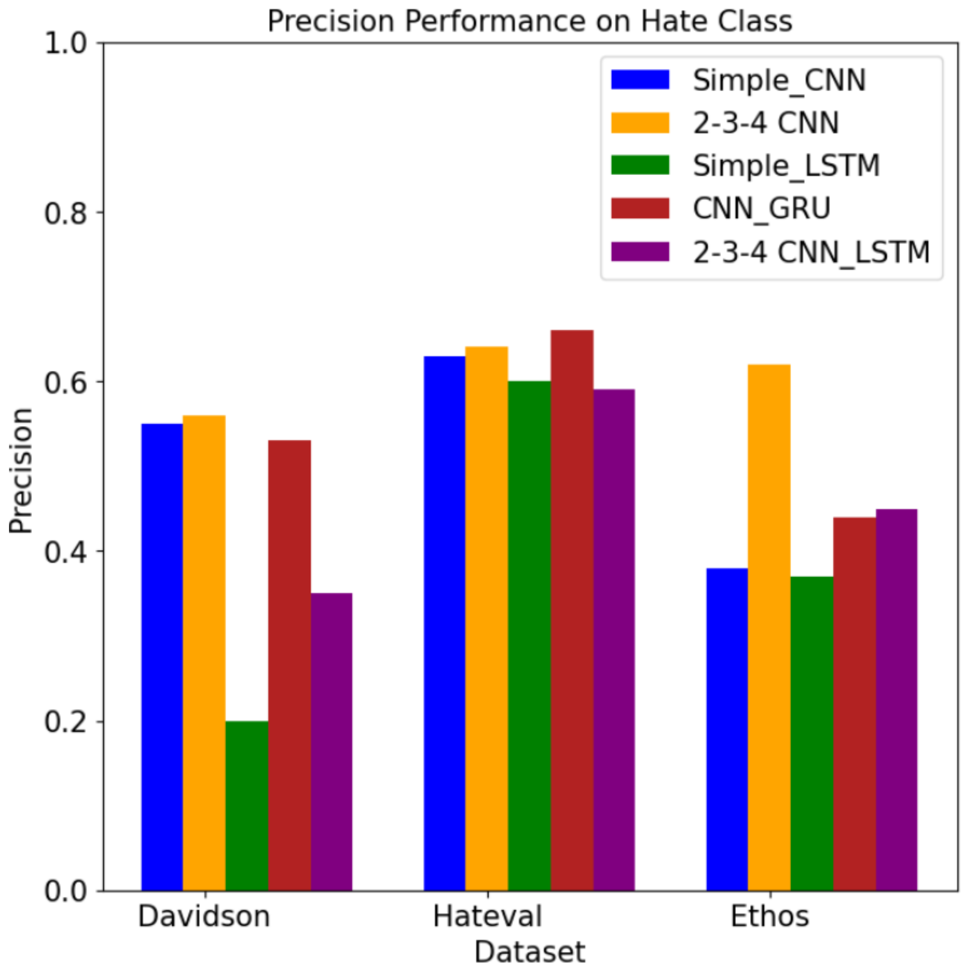

Figure 30, Appendix C-5: Precision Performance on Hate Class of Downstream Models

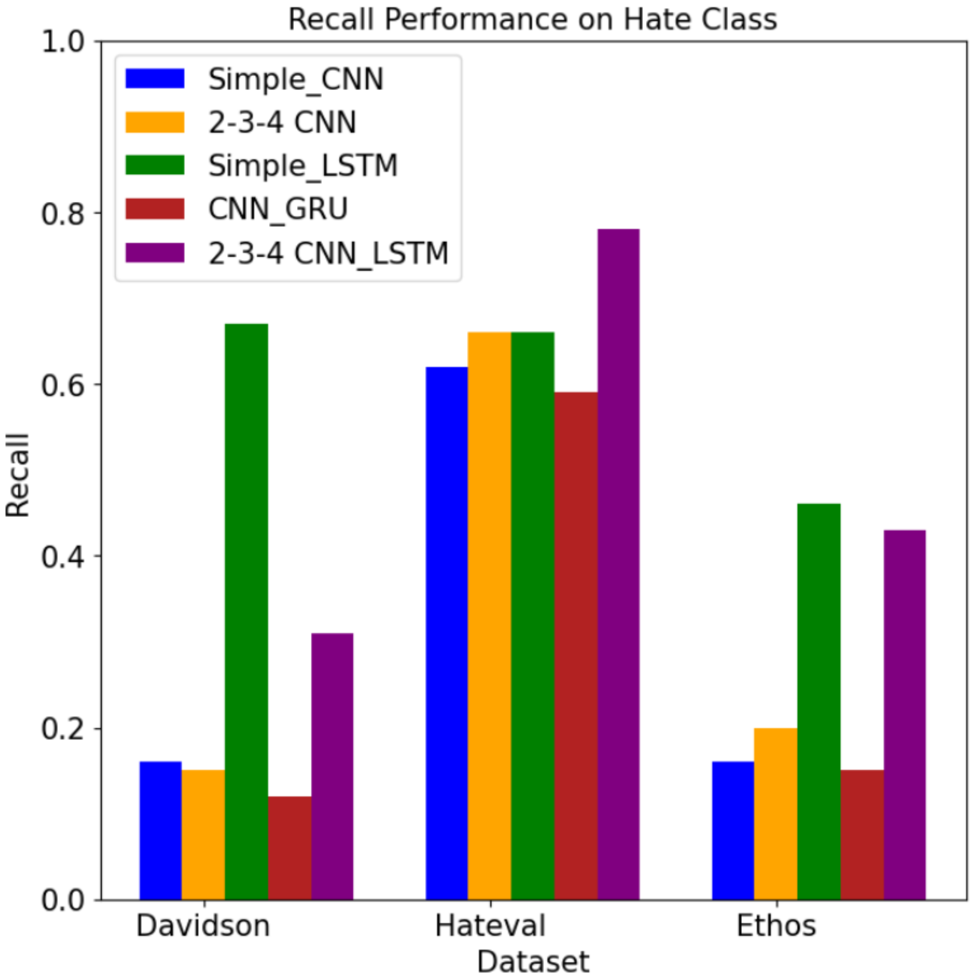

Figure 31, Appendix C-6: Recall Performance on Hate Class of Downstream Models

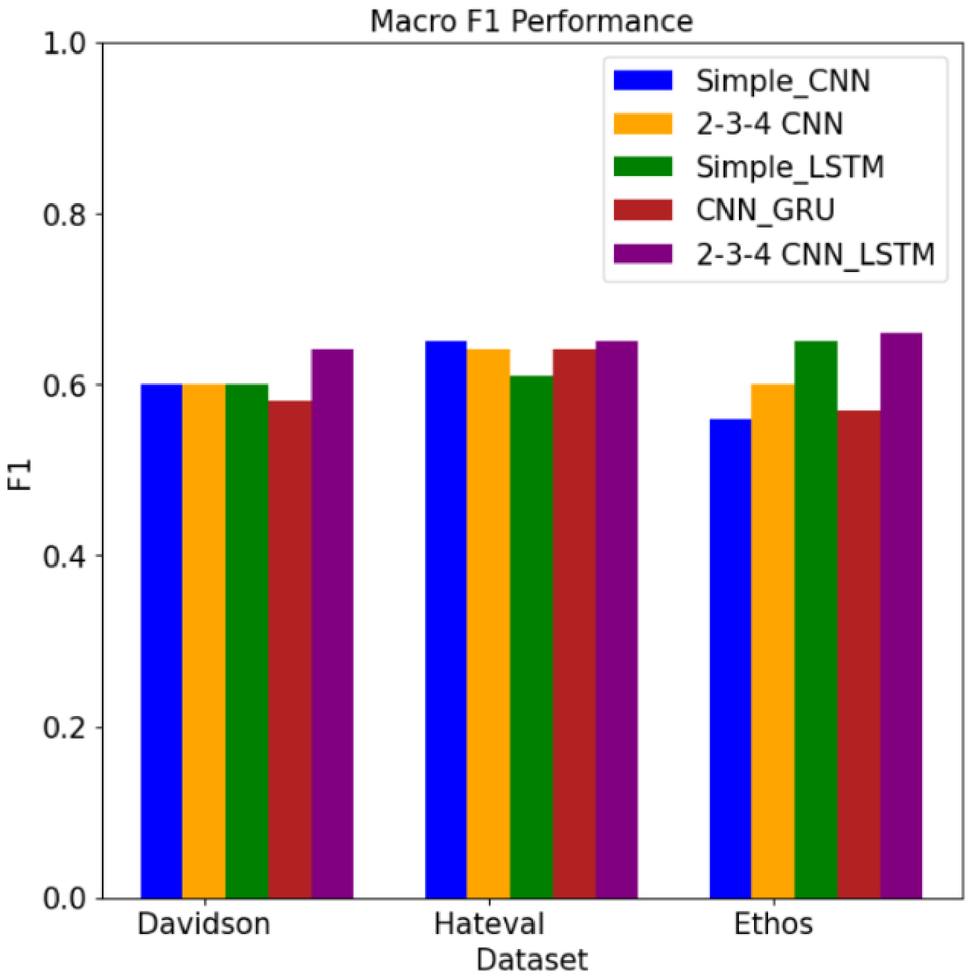

Figure 32, Appendix C-7: Macro F1 Performance of Downstream Models

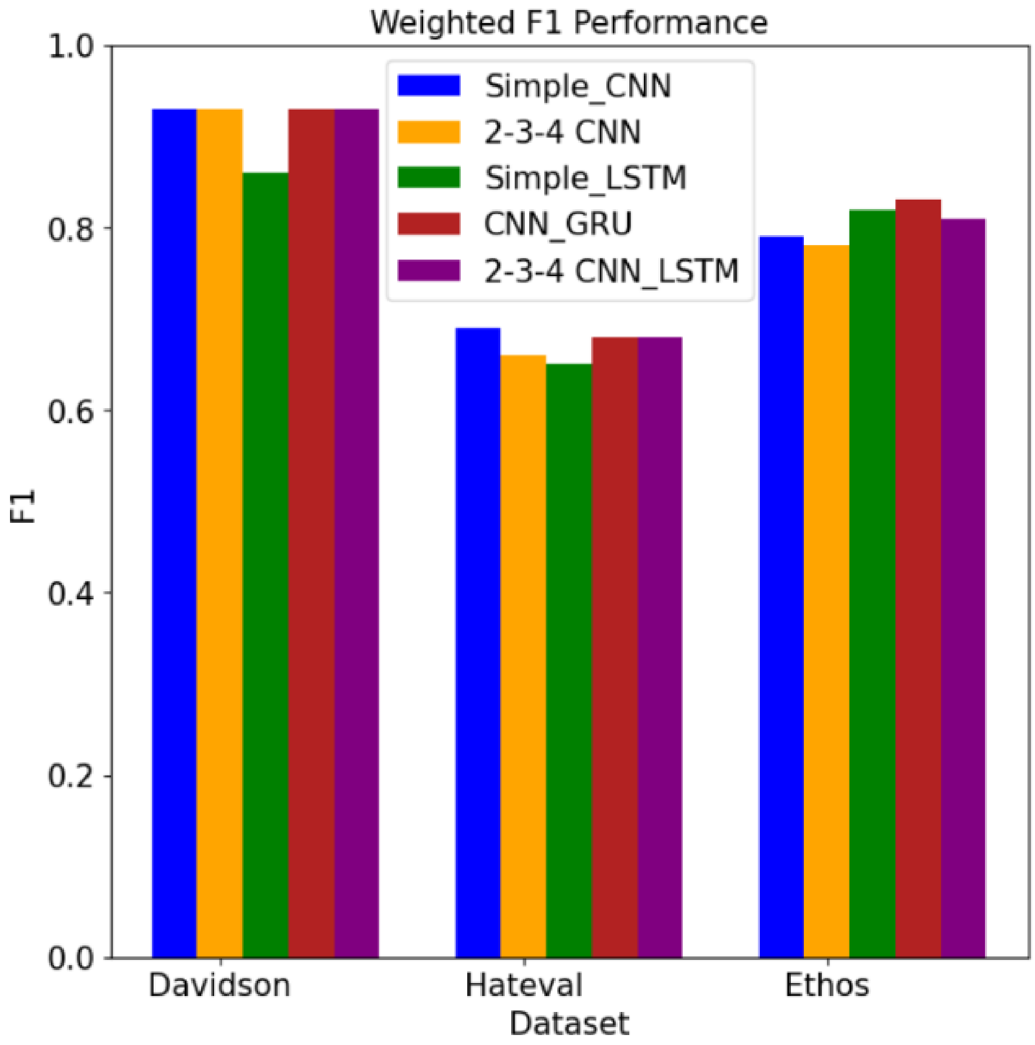

Figure 33, Appendix C-8: Weighted F1 Performance of Downstream Models

D – Github Link

Link: https://github.com/ryu57/EnsembleHateDetection

References

Akhtar, M. S., Ekbal, A., & Cambria, E. (2020). How Intense Are You? Predicting Intensities of Emotions and Sentiments using Stacked Ensemble [Application Notes]. *IEEE Computational Intelligence Magazine*, *15*(1), 64–75. IEEE Computational Intelligence Magazine. https://doi.org/10.1109/MCI.2019.2954667

Ali, R., Farooq, U., Arshad, U., Shahzad, W., & Beg, M. O. (2022). Hate speech detection on Twitter using transfer learning. *Computer Speech & Language*, *74*, 101365. https://doi.org/10.1016/j.csl.2022.101365

Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Rangel Pardo, F. M., Rosso, P., & Sanguinetti, M. (2019). SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter. In J. May, E. Shutova, A. Herbelot, X. Zhu, M. Apidianaki, & S. M. Mohammad (Eds.), *Proceedings of the 13th International Workshop on Semantic Evaluation* (pp. 54–63). Association for Computational Linguistics. https://doi.org/10.18653/v1/S19-2007

cjadams, W. C., Jeffrey Sorensen, Julia Elliott, Lucas Dixon, Mark McDonald, nithum. (2017). *Toxic Comment Classification Challenge*. Kaggle. https://kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). *Automated Hate Speech Detection and the Problem of Offensive Language* (arXiv:1703.04009). arXiv. https://doi.org/10.48550/arXiv.1703.04009

Fortuna, P., Soler-Company, J., & Wanner, L. (2021). How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management*, *58*(3), 102524. https://doi.org/10.1016/j.ipm.2021.102524

Jahan, M. S., & Oussalah, M. (2023). A systematic review of hate speech automatic detection using natural language processing. *Neurocomputing*, *546*, 126232. https://doi.org/10.1016/j.neucom.2023.126232

Mazari, A. C., Boudoukhani, N., & Djeffal, A. (2024). BERT-based ensemble learning for multi-aspect hate speech detection. *Cluster Computing*, *27*(1), 325–339. https://doi.org/10.1007/s10586-022-03956-x

Mollas, I., Chrysopoulou, Z., Karlos, S., & Tsoumakas, G. (2022). ETHOS: A multi-label hate speech detection dataset. *Complex & Intelligent Systems*, *8*(6), 4663–4678. https://doi.org/10.1007/s40747-021-00608-2

Mullah, N. S., & Zainon, W. M. N. W. (2021). Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review. *IEEE Access*, *9*, 88364–88376. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3089515

Qian, J., Bethke, A., Liu, Y., Belding, E., & Wang, W. Y. (2019). *A Benchmark Dataset for Learning to Intervene in Online Hate Speech* (arXiv:1909.04251). arXiv. https://doi.org/10.48550/arXiv.1909.04251

Roy, P. K., Tripathy, A. K., Das, T. K., & Gao, X.-Z. (2020). A Framework for Hate Speech Detection Using Deep Convolutional Neural Network. *IEEE Access*, *8*, 204951–204962. IEEE Access. https://doi.org/10.1109/ACCESS.2020.3037073

Yin, W., & Zubiaga, A. (2021). Towards generalisable hate speech detection: A review on obstacles and solutions. *PeerJ Computer Science*, *7*, e598. https://doi.org/10.7717/peerj-cs.598

Zhang, Z., & Luo, L. (2018). *Hate Speech Detection: A Solved Problem? The Challenging Case of Long Tail on Twitter* (arXiv:1803.03662). arXiv. http://arxiv.org/abs/1803.03662

Zhang, Z., Robinson, D., & Tepper, J. (2018). Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network. In A. Gangemi, R. Navigli, M.-E. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, & M. Alam (Eds.), *The Semantic Web* (pp. 745–760). Springer International Publishing. https://doi.org/10.1007/978-3-319-93417-4_48