HATEXPLAIN SPACE MODEL: FUSING ROBUSTNESS WITH EXPLAINABILITY IN HATE SPEECH ANALYSIS

by

VALLABHANI SIVAMANI RAMA KRISHNA *421269*

MAGGLA PRADEEP *421212*

MADHAV KRISHNA *421211*

Under the guidance of

Dr.K.HIMABINDU

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

TADEPALLIGUDEM-534101, INDIA

MAY 2024

leaving second page

HATEXPLAIN SPACE MODEL: FUSING ROBUSTNESS WITH EXPLAINABILITY IN HATE SPEECH ANALYSIS

Thesis submitted to
National Institute of Technology Andhra Pradesh
for the award of the degree

of

Bachelor of Technology

by

VALLABHANI SIVAMANI RAMA KRISHNA *421269*

MAGGLA PRADEEP *421212*

MADHAV KRISHNA *421211*

Under the guidance of

Dr.K.HIMABINDU

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

TADEPALLIGUDEM-534101, INDIA

MAY 2024

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

V. SIVAMANI RAMA KRISHNA	M. PRADEEP	MADHAV KRISHNA
421269	421212	421211
Date	Date:	Date:

CERTIFICATE

It is certified that the work contained in the thesis titled **"HateXplain Space Model: Fusing Robustness with Explainability in Hate Speech Analysis"** by MADHAV KRISHNA, bearing Roll No: 421211 and MAGGALA PRADEEP, bearing Roll No: 421212, and VALLABHANI SIVAMANI RAMA KRISHNA, bearing Roll No: 421269, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature
Dr.K.HIMABINDU
Assistant Professor & HOD
N.I.T. Andhra Pradesh
MAY 2024

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that I have now the opportunity to express my gratitude for all of them.We owe our sincere gratitude to our project guide Dr. HIMA BINDU K, Department of Computer Science, National Institute of Technology, Andhra Pradesh, who took keen interest and guided us all along, till the completion of our project work by providing all the necessary information and referred many websites .We avail ourselves of this proud privilege to express our gratitude to all the faculty of the department of Computer Science and Engineering at NIT Andhra Pradesh for emphasizing and providing us with all the necessary facilities throughout the work.We offer our sincere thanks to all our fellow mates and other persons who knowingly or unknowingly helped us to complete this project.

ABSTRACT

The identification of online hate speech is a challenging task due to the nuanced and evolving nature of harmful language. Modern Natural Language Processing(NLP) Language Models performs better in various tasks but performs poorly in identifying hate speech while considering zero-short or transfer learning issues. Traditional ”black box” models, while offering some success, lack transparency in their decision-making processes. To address these issues In this project, we implemented a framework called Space Modelling(SM) [1][1] which uses the projection of sentence representations onto task-specific conceptual spaces for improved explainability. Each concept space corresponds to a classification class and is learned in the training phase. This framework shows that each dimension of a class space represents words of a class space. To optimize these class spaces we have used inter and intra space losses. our experiments across 2 hate speech datasets evidences that SM's out performs existing methods. Experimental results demonstrate that the implemented approach enhances detection performance compared to baselines and generates meaningful explanations.

Contents

1 Introduction

Recently, there is a lot of concern about the rise of hateful and offensive language on social media platforms like twitter. This kind of content not only goes against the rules of online platforms but also creates harmful environment on social media. To resolve this issue, researchers and policymakers are working hard to develop ways for computers to automatically recognize and categorize hate speech.

To train models to recognize hate speech, researchers have put some efforts on large sets of examples that include hate speech, offensive language, and normal speech. These datasets are like training datasets for machine learning models, helping them to learn and identify inappropriate content online. But it is not easy because hate speech can be very limited and depends a lot on the context in which it is used.

Researchers have been trying out with different methodoligies to build models that detect the hate speech, from older machine learning techniques to newer deep learning model techniques. One interesting approach is pre-trained language models like BERT, RoBERTa, and XLNet. These models are able to understand the context of words in a sentence better, which helps them to identify the hate speech more accurately.

Not only making accurate models, researchers are also working on making them easier to understand the context. By looking at how these models pay attention to different parts of a sentence and which words they think are most important and which words are not important, researchers can find why the model makes certain decisions. This can help to find the biases or weaknesses in the model's thinking and guide efforts to fix them.

In our project, we implemented different ways of detecting hate speech, methods where the model has not been directly trained on hate speech data (zero-shot classification), ones where it has been trained (supervised fine-tuning). We tested these methods on various hate speech datasets to see which ones are best at finding and sorting offensive content in the context. Additionally, we look more closely at how these models make decisions, analyzing things which parts of a sentence they focus on, to understand how they work and how they could able to improve.

Overall, our project aims to help us in improving the way we detect hate speech online. By better understanding how hate speech detection works and making the models easier to interpret the things, we hope we will contribute to more effective and fair ways of moderating online content in today's digital world.

2 Literature Survey

The process of finding better ways to fine-tuning LLMs has been a prominent topic in the field of Natural Language Processing. Many researches have been conducted to address the challenge of detecting hate speech. Lee et al. proposed the MixOut technique, that mitigates the catastrophic forgetting problem during finetuning LLMs. Instead of updating all the weight of model in every iteration, MixOut randomly selects a small sample of LLMs and updates these weights of the specific sample. Similarly, Xu et al. Proposed a technique called Child Training, which uses the Fisher Information Matrix to identify the target-specific part of the Large Language Models in finetuning, rather than randomly selecting portion of the LLMs. another recent work in the area is CAMERO(Liang et al) where we use weight sharing technique between the bottom layers of models and different Turbulences were applied to the hidden representations.

In recent years, many datasets have been published for the detection of hate speech tasks. Wasseem and Hovy created a dataset that points on disambiguating racism.This dataset has 16,914 tweets, with 3,383 labels as sexist,1,972 as racist, and 11,559 as neither. Davidson et al. published a dataset for differentiating tweets between hate speech, offensive language, and normal speech.This dataset contains 25,000 tweets from twitter collected from twitter API, with 77% of offensive language,6% of hateful language and rest of tweets are categorized as normal. Founta et al. introduced a dataset having 80,000 tweets containing abusive, hateful, normal, and spam content. More recent datasets has been collected from various sources and incorporated diverse criteria for hate speech classification. Zampieri et al. created the Offensive Language Identification Dataset(OLID),which is considered to be one of the finest datasets for hate speech detection containing 14,000 tweets where 4,500 tweets labeled as offensive. This dataset employs a three-level hierarchy annotation schema that includes hate speech detection, categorization, and target identification. kiela et al. published a dataset of 40,000 entities, with 54% of them labeled as hateful in which they employed binary labeling schema helps to identify the type and target of hate speech. Many annotators provided feedback for the annotation process of the dataset. The HateXplain dataset is the most recent dataset in hate speech detection datasets and includes human rationales.

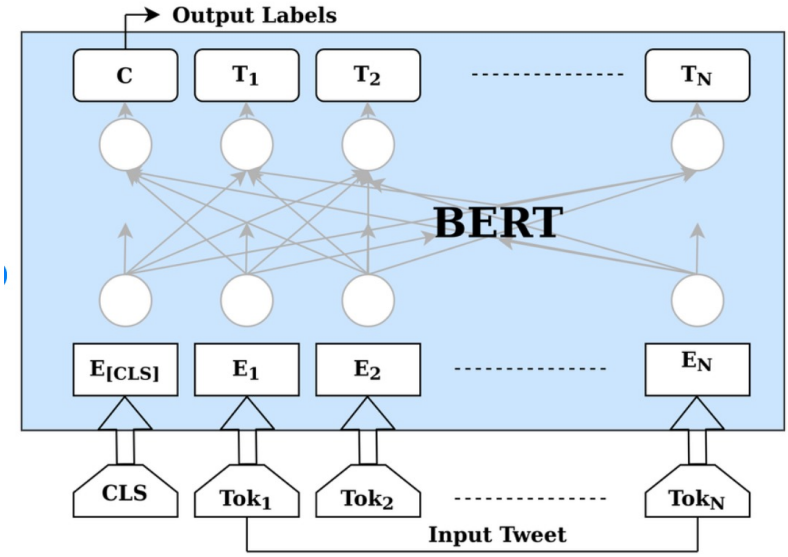


figure1:This figure gives the overview of the classic BERT model. Which takes sequence of input tokens of maximum length 512 it has 12 layers of transformer blocks in BERT-base and 24 in BERT-large. It gives output tokens of 768 dim for BERT-base and 1024 for BERT-large. It generates CLS token which is used as input for feed forward network in classification tasks.(ResearchGate)

Wide range of machine learning models were used for hate speech classification which includes logistic regression, decision tress, random forests, k-means etc. Some processes incorporate word embeddings such as Word2Vec (Mikolov et al.), FastText(pennington et.al) to obtain vector embedding of words, in combination with neural architectures like CNNN(Zhang Wallace),LSTM, and GRUU (Chung et al.).Currently, transformers based Large Language models such as BERT and variations of BERT are being used these Transformers based approaches outperformed other models in NLP tasks including hate speech detection.Unlike previous models that processed text in one direction BERT utilizes a bidirectional approach, which allows it to consider both preceding and following context when encoding words. This bidirectional encoding enables BERT to better understand the context and meaning of words in a sentence, enabling to more accurate representations of language.BERT is pre-trained on large amounts of text data using unsupervised learning, where it learns to predict missing words in sentences based on the surrounding context. This pre-training process allows BERT to capture rich semantic information and syntactic structures from the input text. Work by Caselli et al. introduced HateBERT T (Caselli et al.) a pretrained BERT model for hate speech identification in English. The HateBERT was trained on large scale dataset of Reddit comments in English from communities banned for being offensive, abusive and hateful.In all cases, HateBert outperformed existing BERT model.

In conclusion, hate speech detection is a challenging yet essential task in NLP, with ongoing research focusing on improving model performance, developing robust techniques, and creating annotated datasets to address the complex nuances of harmful language online.

3 Data Sets

The dataset compilation for hate speech analysis encompasses several significant contributions from various sources:

3.1 Dataset Description

Several datasets have been compiled and released to address the challenge of discerning hate speech, offensive language, and normal speech within social media content, particularly Twitter. Researchers have undertaken efforts to construct comprehensive datasets, each with its own unique characteristics and annotation schema.

One such dataset, comprising 25,000 tweets collected through the Twitter API, was released to classify content into hate speech, offensive language, and normal speech categories. Another dataset, consisting of 80,000 tweets, categorized content as abusive, hateful, normal, or spam. Moreover, OLID dataset stands out as one of the most established datasets, containing 14,000 tweets, with 4,500 labeled as offensive. OLID employs a three level annotation for hate speech detection, categorization, and target identification.

In addition, a dataset generated through a human-and-model-in-the-loop process comprised 40,000 entries, with 54 percent identified as hateful. This dataset identify the type and target of hate speech. Furthermore, another dataset introduced by researchers contained 10,000 texts, with approximately 11 percent labeled as hate. Moreover, a dataset of implicit hate speech, collected from Twitter, included around 20,000 samples, with approximately 5,000 being implicit hate samples.

To ensure consistency in labeling across different datasets, efforts were made to unify class labels. Some datasets utilized three class labels, while others used two. To address this discrepancy, the hate and offensive classes were merged into a single hate class. Subsequently, normal text was considered as the non-hate class, and the merged hate class encompassed both hate speech and offensive language. This conversion facilitated consistency in dataset labeling and enabled researchers to analyze and compare datasets more effectively.

Overall, the compilation and standardization of these datasets have significantly contributed to research in hate speech detection and mitigation. By leveraging diverse datasets with unified labeling, researchers can develop and evaluate models and algorithms aimed at combating hate speech and promoting a safer online environment.

Dataset	Train Data(Non Hate)	Train Data(Hate)	Test Data(Non Hate)	Test Data(Hate)
OLID	7107	3485	1733	915
Davidson	3328	16498	835	4122

3.2 Data Cleaning

In this code segment, the initial focus lies on loading and preparing the dataset for hate speech classification using a BERT-based model. The dataset, stored in a CSV file, is loaded using the pandas library, which allows easy manipulation and analysis of tabular data in Python. Once loaded, the dataset is structured into a DataFrame, where each row represents a tweet along with its corresponding labels and other relevant information. Before proceeding further, the class labels are transformed into a binary format to distinguish between hate speech and non-hate speech instances. This conversion simplifies the classification task by collapsing the original multi-class labels into two categories: hate speech and not-hate speech.

Moving on to the actual tweet text, a cleaning process is initiated to extract only the essential content from each tweet while discarding any extraneous information such as usernames or metadata. This step is crucial for focusing solely on the textual content of the tweets, which is pertinent for hate speech classification. The cleaning process involves splitting each tweet based on the colon (':') character, as it often separates the actual tweet content from other accompanying details. If a tweet contains a colon, only the last part, representing the actual tweet text, is retained; otherwise, the entire tweet is kept intact. This ensures that the tweet data is streamlined and ready for further processing, including tokenization. Tokenization, performed using the BERT tokenizer, converts the cleaned tweet

text into a format suitable for input into the BERT model, laying the groundwork for subsequent model training and evaluation. Overall, these meticulous data preprocessing steps are crucial for ensuring the quality and integrity of the dataset, ultimately contributing to the effectiveness and reliability of the hate speech classification model.

3.3 Creating Hate and Non-Hate Spaces

To create a predefined hate space, we selected 11 words known to be commonly associated with hate speech. These words were chosen based on various sources ranking them as highly offensive or hateful. Examples include ”Moist,” ”Cunt,” ”Panties,” ”Fuck,” and others. We then passed these individual words through pre-trained language models like BERT to extract their embeddings, which are numerical representations of the words’ meanings. By combining these embeddings, we formed a hate space, essentially a mathematical representation of hate-related language.

Similarly, to define a non-hate space, we selected 11 words that are essentially opposite in meaning to the hate words. These words, like ”Love,” ”Peace,” ”Kindness,” and ”Happiness,” were chosen to represent positive sentiments and values. We extracted embeddings for these words from language models and combined them to create a non-hate space. Essentially, these spaces serve as reference points or representations of different types of language, allowing us to analyze and understand text data in terms of hate speech and non-hate speech.
These are the 11 words selected for hate space and not hate space respectively.

[’Moist’, ’Cunt’, ’Panties’, ’Fuck’, ’Hate’, ’Nigger’, ’Pussy’, ’Ass’,’Motherfucker’, ’Bitch’, ’Damn’] [’Love’, ’Peace’, ’Kindness’, ’Happiness’, ’Respect’, ’Friendship’, ’Appreciation’, ’Hope’, ’Encouragement’, ’Support’, ’Caring’]

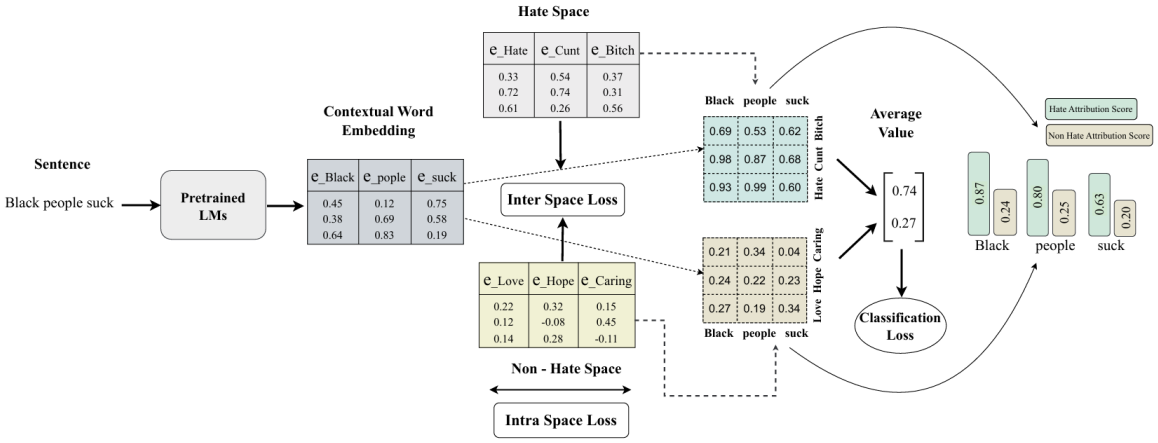


figure2:Architecture of Space-Model.for a sample sentence we find word level attribution scores for hate and not hate spaces by projecting vector embeddings of sentence onto the class space embeddings which enhances the model explainability.(Md.Fahim et al.)

4 Methodology

figure(i) shows the model architecture of space model for a sample sentence. If we consider a classification task of c classes. For a sentence s, we pass it to a fine-tuned BERT model we will find the contextual embeddings of each word from the last hidden layer representation $E = [e_1, e_2, ..., e_n]$, where n represents the number of words in the sentence and d denotes the dimensionality of the embeddings. For each class we consider m different words which mostly define the class. we pass these embeddings to fine-tuned BERT model to get vector embeddings of these words then we concatenate embeddings of each class space words in such a way that we obtain class space $S_k \in R^{d \times m}$ for k$^{\text{th}}$ class.similarly for each class we will find class space $S_0, S_1, ..., S_{c-1}$.

4.1 Fine-tuning BERT

we fine-tuned BERT model on downstream task for respective datasets by adding two additional feed-forward layers and a softmax layers on top of cls token for k class classification.we finetuned BERT model for this classification task by updating model parameters minimally with very low learning rates in the order of e^{-5} for certain number of epochs.

4.2 Space Model

we project contextual embeddings obtained from BERT model onto the class spaces using cosine similarity by finding cosine similarity between each embedding and each word from space which results in a matrix.For a class K we project embeddings of a word e_i onto the class space.

$$p_{i,Sk} = \frac{e_i^T \cdot Sk}{||e_i|| \cdot ||Sk||}$$

$E_s \in R^{d \times n}$, E_s are the set of embeddings of the words of a sentence s with n words each embedding having d dimensions. we project these embeddings matrix on to the class space S_k to get a matrix $S_k \in R^{m \times n}$. each element of this psk matrix defines the cosine similarity between word of a sentence i and word from class space k. For each word in the sentense we can get attribution score by finding column wise average of the $P_{S,k}$ matrix.

The authores defined two types of space models i)Supervised Space Model(SVM) and ii)Semi-supervised Space Model(Semi-SSM).

SSM doest require any training we it resumbels ZERO-SHORT text classifiaction.For a sentence we simply pass it to BERT and find cosine similarity with the class spaces.The class coressponding to maximum cosine similarity is considerd as sentece class.

In Semi-SSM we don't fix these embeddings we will make them trainable along with weights associated with each class space.For training authors introduced 2 additional losses Inter and Intra space losses along with classification loss(Cross Entropy).

For prediction in Semi-SSM we pass each setence throgh BERT model to obtain contextual embeddings we project these embeddings on to the class space and we will take the row wise average and takes weighted sum of these row wise averages as sentence level attribution score for each class space we assign class based on maximum score of a class.This semi-SSM resumbels transfer learning approch.

4.3 Inter and Intra space loss

In semi-SSM,Authors have introduced inter-space loss and Intra-space loss.

The job of inter-space loss is to ensure that the embeddings of the two different classes are orthogonally apart form each other.Linter enforeses the model to learn disjoint class spaces.for k[th] class they calculated mean of the S_k matrix denoted as μ_k then they calculated the sum of inter-space loss for every pair of class spaces as the total inter-space loss.

$$L_{inter} = \sum_{k=0}^{c-1} \sum_{l=0;l\neq k}^{c-1} \frac{1}{1 - \mu_k \mu_l / ||\mu_k|| \cdot ||\mu_l||}$$

To ensure the embeddings within the class don't converge to the same word word embedding ,They introduced Intra-space loss.The loss L_intra for S_k concept space for K^{th} class enforces the concept space embeddings to be different which intern helps us to capture broad information when we use cosine similarity. $Var(S_k) = \frac{1}{m}\sum_{i=1}^{m}(w_i - \tilde{w})^2$, where w_i represents the i^{th} word embedding of the class space matrix S_k and $\tilde{w}$ is the column wise average of the class space S_k.The inter space loss for S_k is defined as

$$L_{intra,Sk} = \frac{1}{Var(S_k)}$$

the total loss accross all the class spaces is defined as

$$L_{intra} = \sum_{n=0}^{c-1} L_{intra,S_k}$$

the new loss functions has 3 components :

$$L = L_{CE} + \lambda_1 L_{inter} + \lambda_2 L_{intra}$$

we train the model to minimise this effective loss function.

By integrating the Space Model architecture with SSM and Semi-SSM variants, the hate speech detection project achieves robust classification performance. The model effectively leverages the contextual information captured by the LM embeddings and optimizes the class space embeddings to accurately classify hate speech within text data. The incorporation of inter-space and intra-space losses in Semi-SSM further enhances the model's capability to learn meaningful representations for hate speech classification.

5 Experiments and Results

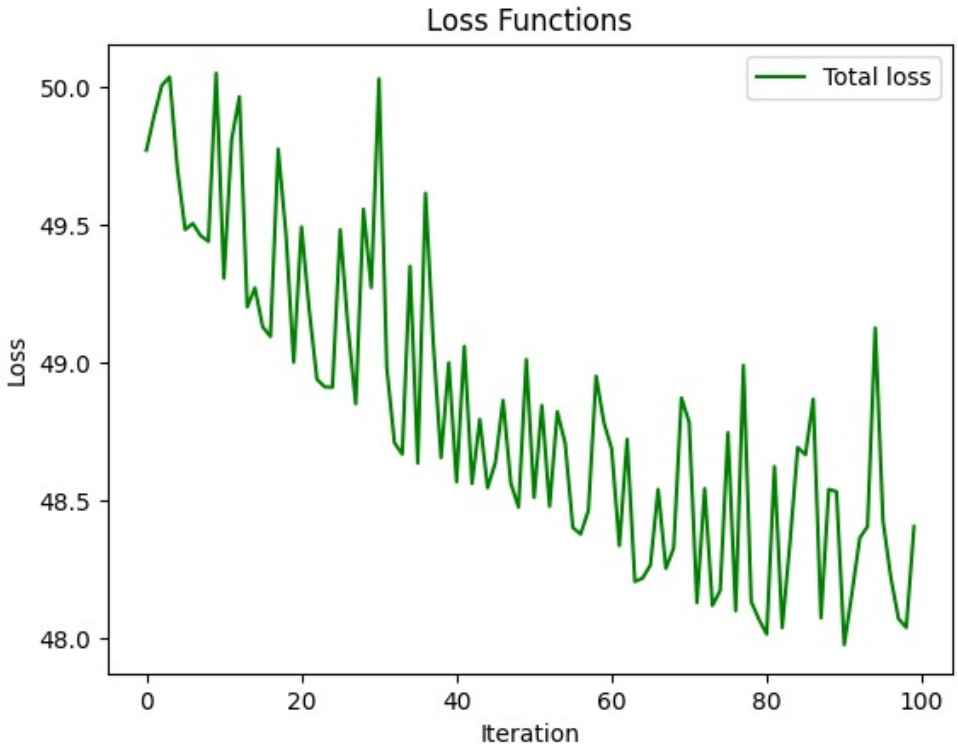
5.1 Experiments

5.1.1 Finetuning Bert Model

Finetuning is very popular when we are dealing with BERT model. Finetuned BERT model performs better when compared to only Pretrained BERT model by significant margin. In our project we finetuned BERT model on downstream task with training data by adding two feed forward layers with ReLu activation function and softmax layer on top we finetuned BERT model for three since running for more number of epochs could cause overfitting problem epochs with very small learning rate which only updates weights of the BERT model by very tiny amount we take small learning rate since taking larger values of learning rates could potentially disturb pretrined weights in finetuning we don't want to disturb those we try to change them a little bit. From our observations we have witnessed clear spike in the accuracy scores of the model on different datasets we have seen nearly 5% improvement in accracy for Davidson dataset and 10% rise in accuracy for OLID dataset. Which shows the power and necessity of finetuning.

5.1.2 Tuning hyper parameters

Since we have three losses in SM model it is essential for us to find correct mixture of these losses to get optimum results we should maintain the significance of crossentrophy loss while maintaing some importance to the inter and intra space losses due to the problem of anisotropy the BERT embeddings are almost similar for any words due to this problem the Inter and Intra space losses spikes up to prevent them from dominating classification loss we have found from experiments that 0.1 value for the hyper parameters corresponding to space losses yields better results .



Graph1:This Graph shows the loss function values corresponding to no.of iterations under the hyper parameters of 0.1 for space losses with a learning rate of 0.002.

5.2 Performances and Test Accuracy

Dataset	Model	Accuracy	Macro Precision	Macro Recall
OLID	Zero Shot(SOTA)	54.79	46.41	45.97
	SSM	64.51	63.59	64.96
	Transfer Learning(SOTA)	66.96	67.87	53.04
	Semi-SSM	77.79	75.8	73.83
Davidson	Zero Shot(SOTA)	64.34	50.05	50.44
	SSM	75.15	62.4	68.35
	Transfer Learning(SOTA)	87.14	83.05	65.49
	Semi-SSM	92.27	88.1	82.88

Table 1: Table1:Model performance of Semi-SSM, SSM for 2 different datasets are compared against ZERO-SHORT and Transfer Learning. The SSM outperforms ZERO-SHORT model by significant margin.The Semi-SSM outperforms rest of the models by significantly large margin.

The table presents performance metrics for different models on two datasets: OLID and Davidson. Each model is evaluated based on accuracy, macro precision, and macro recall.

For the OLID dataset, the "Zero Shot" model achieves an accuracy of 54.09%, with macro precision and macro recall scores of 46.01% and 45.7% respectively. The "SSM" (Single Stage Model) performs better with an accuracy of 63.61% and slightly higher precision and recall scores at 63.5% and 63.6%. The "Semi-SSM" (Semi-Supervised Model) tops the chart with an accuracy of 77.79%, exhibiting significantly improved macro precision (75.8%) and macro recall (73.83%).

Switching to the Davidson dataset, the "Zero Shot" model obtains an accuracy of 64.34%, with macro precision and macro recall standing at 50.05% and 50.44%. The "SSM" model enhances performance further, achieving an accuracy of 75.55% with macro precision and macro recall scores of 62.4% and 68.10% respectively. Finally, the "Semi-SSM" model showcases exceptional performance with an accuracy of 92.27%, along with notably high macro precision (88.1%) and macro recall (82.88%) scores.

5.3 Performance Analysis

Table 1 compares the performance of the SSM and Semi-SSM models to state-of-the-art (SOTA) models. The SSM significantly outperforms traditional zero-shot models by a large margin of 10-20%, while the Semi-SSM surpasses transfer learning approaches by 5-15%.

In zero-shot text classification, language models (LMs) are directly tested on new data without any prior training. Similarly, the Single Stage Model (SSM) doesn't need training and is directly evaluated on test data. For transfer learning, a classification head is added on top of a BERT model with its parameters frozen, and then this classification head is trained. On the other hand, the Semi-SSM also involves training the weights of the Space Model but keeps the BERT model parameters frozen.

Looking at each dataset in Table 1, we find that our SSM model performs better than zero-shot techniques, improving by 10-20% across different metrics. For the Semi-SSM, it outperforms transfer learning techniques for four out of six datasets by a considerable margin. However, it slightly falls short for the other two datasets. This difference may be due to the words chosen to represent the classes not being the best representatives. Additionally, we are also exploring the impact of losses and different language models, which are detailed in Appendix B.1 for loss effects and Appendix B.2 for the effects of choosing different LMs.

6 Future Work

Enhanced Model Architecture: Future work could focus on developing the model architecture to improve its performance more. This involves experimenting with different neural network architectures, exploring more advanced techniques such as self attention mechanisms , or integrating external knowledge sources to improve model understanding.

Exploring Additional Datasets: While this study experimented with several datasets, there are number of other hate speech datasets are also available. Future work involves evaluating the model on additional datasets to assess its generalization across different domains and languages.

Multi modal Hate Speech Detection: Integrating multiple things such as text, images, and audio could provide a more comprehensive understanding of hate speech. Future work could explore multi modal approaches to improve hate speech detection accuracy and stability.

7 Conclusion

In this project, we implemented a novel based approach for hate speech detection framework based on semi-supervised learning and space modeling(Md.Fahim et al.). Through extensive experimentation on multiple datasets, we performed the effectiveness of our approach in accurately identifying hate speech while helping to minimize false positives. Our results indicate that our model performs well than baseline methods, including zero-shot classification and transfer learning, across various performance metrics we have performed.

Additionally, we conducted explainability analysis to gain insights into the model's decision-making process, providing interpretable explanations for hate speech detection in the context. This transparency improves the trust in our model and enables stakeholders to understand and validate its predictions.

Overall, our study focus on the growing body of research aimed at detecting hate speech online. By developing robust and interpretable hate speech detection models, we developed the way for safer and more friendly online communities.

8 References

[1]HateXplain Space Model: Fusing Robustness with Explainability in Hate Speech Analysis by MdFahim et al.

[2]Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In Proceedings of the international AAAI conference on web and social media, volume 11, pages 512–515, 2017.

[3]Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. Hate speech dataset from a white supremacy forum. In Proceedings of the 2nd Workshop on Abusive Language Online (ALW2), pages 11–20, Brussels, Belgium, October 2018. Association for Computational Linguistics.

[4]Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[5]Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. Latent hatred: A benchmark for understanding implicit hate speech. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 345–363, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[6]Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. arXiv preprint arXiv:1909.00512, 2019.

[7]Paula Fortuna, Monica Dominguez, Leo Wanner, and Zeerak Talat. Directions for NLP practices applied to online hate speech detection. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11794–11805, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[8]Antigoni Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. Large scale crowdsourcing and characterization of twitter abusive behavior. In Twelfth International AAAI Conference on Web and Social Media, 2018.

[9]Goran Glavaš, Mladen Karan, and Ivan Vulić. XHate-999: Analyzing and detecting abusive language across domains and languages. In Proceedings of the 28th International Conference on Computational Linguistics, pages 6350–6365, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[10]Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. arXiv preprint arXiv:2111.09543, 2021.

[11]Shaoyi Huang, Dongkuan Xu, Ian EH Yen, Yijue Wang, Sung-En Chang, Bingbing Li, Shiyang Chen, Mimi Xie, Sanguthevar Rajasekaran, Hang Liu, et al. Sparse progressive distillation: Resolving overfitting under pretrain-and-finetune paradigm. arXiv preprint arXiv:2110.08190, 2021.

9 Appendix

9.1 Model Interpretability

While deep learning models like BERT and RoBERTa achieve the performance in NLP tasks, they are often considered black-box models due to their complex architectures. Interpreting the decisions made by these models, especially in sensitive tasks like hate speech detection, poses a great challenge.

9.2 Zero-Shot Learning

Traditional fine-tuning approaches require labeled hate speech data for each target domain, which may not always be available. Zero-shot learning aims to address this limitation by enabling models to generalize to unseen domains without explicit training. However, adapting zero-shot learning techniques to effectively detect hate speech remains a challenge.

9.3 Data Annotation

The data annotation process involves several steps aimed at preparing the dataset for hate speech detection using a BERT-based model. Initially, the dataset is loaded from a CSV file using the pandas library, a powerful tool for data manipulation in Python. Each tweet, along with its associated labels indicating hate speech or non-hate speech, is structured into a DataFrame. To facilitate classification, the class labels are transformed into a binary format, simplifying the task by distinguishing between instances of hate speech and those that are not. Subsequently, the actual tweet content undergoes cleaning to extract essential information while discarding extraneous details like usernames or metadata. This cleaning process ensures that only the relevant text data is retained for analysis. Once cleaned, the tweet text is tokenized using the BERT tokenizer, breaking it down into manageable units for processing by the BERT model. Finally, the tokenized text is ready for input into the BERT model for hate speech detection. By meticulously annotating and preprocessing the dataset, the code sets the stage for training and evaluating a robust hate speech detection model, essential for promoting a safer and more inclusive online environment.

9.4 BERT Model

Overview: BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model known for its ability to generate contextual word embeddings from the given input text. It will be achieved this by utilizing self-attention mechanisms and bidirectional context understanding during pre-training on large amounts of text data. Given an input text sequence, BERT tokenizes the text into subwords, which are then converted into embeddings and processed through multiple transformer layers. These transformer layers capture contextual information and produce final hidden states, which serve as contextual word embeddings. These embeddings encode the semantic meaning of each word within the context of the entire sentence.

9.5 Effect of Losses

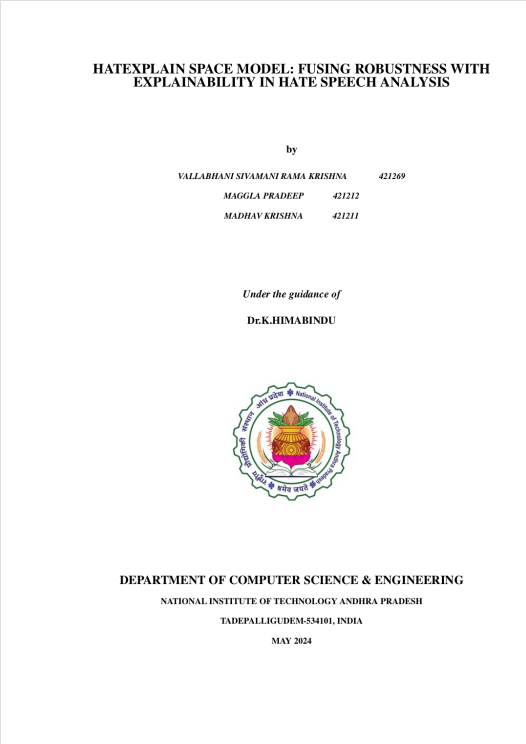
The authors have done experiments on the losses and their effects by considering(OLID,Davidson) datasets.For each dataset the authors excluded inter and intra space losses and found model accuracy. The report that the accuracy were reduced by significant margin 2-5% which concludes that both these losses are essential for Semi-SSM model.

Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	vallabhani sivamani rama krishna
Assignment title:	MiniProjectReport
Submission title:	Project Report
File name:	HateSpeech_Project_Report_16_.pdf
File size:	449.46K
Page count:	20
Word count:	5,647
Character count:	31,581
Submission date:	09-May-2024 10:14AM (UTC+0530)
Submission ID:	2372102545



HATEXPLAIN SPACE MODEL: FUSING ROBUSTNESS WITH EXPLAINABILITY IN HATE SPEECH ANALYSIS

by

VALLABHANI SIVAMANI RAMA KRISHNA *421269*

MAGGLA PRADEEP *421212*

MADHAV KRISHNA *421211*

Under the guidance of

Dr.K.HIMABINDU

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

TADEPALLIGUDEM-534101, INDIA

MAY 2024

Project Report

ORIGINALITY REPORT

12%	10%	5%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	4%
2	nips.cc Internet Source	1%
3	Md Saroar Jahan, Mourad Oussalah. "A systematic review of hate speech automatic detection using natural language processing", Neurocomputing, 2023 Publication	1%
4	dokumen.pub Internet Source	1%
5	aclanthology.org Internet Source	<1%
6	avalon.ira.uka.de Internet Source	<1%
7	dgeconomy.tsue.uz Internet Source	<1%
8	docplayer.net Internet Source	<1%