Towards Interpretable Hate Speech Detection using Large Language Model-extracted Rationales

Ayushi Nirmal[*] **Amrita Bhattacharjee**[*] **Paras Sheth** **Huan Liu**
School of Computing and Augmented Intelligence
Arizona State University
{anirmal1, abhatt43, psheth5, huanliu}@asu.edu

Abstract

Although social media platforms are a prominent arena for users to engage in interpersonal discussions and express opinions, the facade and anonymity offered by social media may allow users to spew hate speech and offensive content. Given the massive scale of such platforms, there arises a need to automatically identify and flag instances of hate speech. Although several hate speech detection methods exist, most of these black-box methods are not interpretable or explainable by design. To address the lack of interpretability, in this paper, we propose to use state-of-the-art Large Language Models (LLMs) to extract features in the form of rationales from the input text, to train a base hate speech classifier, thereby enabling faithful interpretability by design. Our framework effectively combines the textual understanding capabilities of LLMs and the discriminative power of state-of-the-art hate speech classifiers to make these classifiers faithfully interpretable. Our comprehensive evaluation on a variety of English language social media hate speech datasets demonstrate: (1) the goodness of the LLM-extracted rationales, and (2) the surprising retention of detector performance even after training to ensure interpretability. All code and data will be made available at https://github.com/AmritaBh/shield.

1 Introduction

Content Warning: This document contains content that some may find disturbing or offensive, including content that is discriminative, hateful, or violent in nature.

Social media has become a platform of content sharing and discussions for a varied range of individuals with differing cultural and continental backgrounds. People use social media platforms to exchange information, and they frequently engage in dialectal conversations. These discussions are not always peaceful, they can degenerate into unpleasant altercations and bigoted arguments. Thus, social media platforms often become a host for hate speech. Hate speech is described as any deliberate and purposeful public communication meant to disparage a person or a group by expressing hatred, disdain, or contempt based on their social attributes (e.g., gender, race). In extreme cases, hate speech may often lead to real world harms such as hate crimes, for example the anti-Asian hate crimes during the COVID-19 pandemic (Findling et al., 2022; Han et al., 2023). Therefore, it is essential to have automatic hate speech detection and moderation in place to maintain the integrity of social media platforms as well as to mitigate negative impacts in real-world scenarios such as increased violence towards minorities (Laub, 2019).

Given that the issue of hate speech on social media is a well-established problem, there have been several works to detect such online hate-speech (Schmidt and Wiegand, 2017; Del Vigna12 et al., 2017). While state of the art hate speech detection models have been able to achieve good performance on benchmark evaluation datasets, most of these models are built using transformer-based pre-trained language models or other deep neural network type models (Sheth et al., 2023b) that are not interpretable or explainable. However, the task of hate speech detection is a very sensitive task, and explainability of automated detectors is an essential and desirable feature. Model interpretability is essential not only for end-user understanding but also for understanding biased predictions, domain shifts, other errors in the prediction, etc.

While incorporating qualities of interpretability directly into deep neural network models such as pre-trained language model based detectors is challenging, one way to potentially perform this is by

[*] These authors contributed equally to this work.

Proceedings of the The 8th Workshop on Online Abuse and Harms (WOAH), pages 223–233
June 20, 2024 ©2024 Association for Computational Linguistics

using an auxiliary model to provide explanations or rationales, that are subsequently used in training the detection model. This type of a method has been proposed and used in the FRESH framework (Jain et al., 2020), where the authors use two disjoint networks, one for extracting the task-specific rationales, and then another that leverages those rationales to learn the classification task, thereby enabling faithful interpretability *by construction*.

Inspired by this work, we propose a framework, where we use LLMs as the extractor model: we leverage the textual understanding and instruction-following capabilities of state-of-the-art LLMs to extract features from the input text, that is used to augment the training of a separate base hate speech detector, thereby facilitating faithful interpretability. Overall, our contributions in this paper are:

1. We propose **SHIELD**, a framework that leverages LLM-extracted rationales to augment a base hate speech detection model to facilitate faithful interpretability.

2. We evaluate the goodness of LLM-extracted features and rationales, and measure the alignment of such with human annotated rationales.

3. Through comprehensive experiments on both implicit and explicit hate speech datasets, we show how **SHIELD** retains detection performance even after training with rationales for increased interpretability, despite the expected interpretability-accuracy trade-off.

2 Our SHIELD Framework

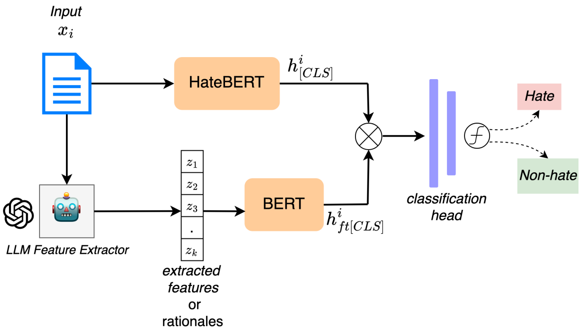


Figure 1: Our proposed **SHIELD** framework.

We show our proposed **SHIELD** framework in Figure 1. In this section, we describe our framework in detail, elaborating on each of the components.

LLM Feature Extractor Our framework uses the state-of-the-art instruction-tuned large language models (LLMs) in an off-the-shelf manner as textual feature extractors. Although recent work has shown that LLMs struggle to perform the hate speech detection task (Li et al., 2023; Zhu et al., 2023) when used without any additional model or fine-tuning, we hypothesize that we can leverage the textual understanding capabilities of these LLMs to simply extract textual features in the form of rationales. Restricting the use of the LLM to a simple text-level task would ensure that such models are not directly being used for sensitive application tasks such as hate speech detection (Harrer, 2023). For a given input text $x_i \in X$, we use our carefully designed task prompt to prompt the LLM to extract features from the text that promotes a hateful sentiment. In the context of explicit hate speech detection, such features could include categories such as derogatory words, cuss words, etc. Following similar work in (Bhattacharjee et al., 2023b), we also ask the LLM for rationales as to why the label is hateful or non-hateful. To perform this feature extraction, for each input text we prompt the LLM using the following prompt:

> “You are a content moderation bot. Identify the list of rationales, list of derogatory language, list of cuss words that promote a hateful sentiment and respond with non-hateful if there are none. Note: The output should be in a json format.”
> Text: [input_text]

After post-processing the outputs, we have a list of k textual features $\{z_j\}_{j=1}^k$ for the given input text x_i.

Hate Speech Detector as Embedding Module The next component in our framework is the base hate speech detector which we are trying to augment, such as HateBERT (Caselli et al., 2020). HateBERT is a BERT (Devlin et al., 2018) model that is specifically fine-tuned on hate speech data. For each input text $x_i \in X$, instead of obtaining the labels or class probabilities, we take the last layer embedding of the `[CLS]` token, $h^i_{[CLS]}$, essentially containing all the information of the input text, that is relevant for the hate-speech detection task.

Feature Embedding Model For the textual features and rationales, $\{z_j\}_{j=1}^k$, we extracted via the LLM, we use a pre-trained transformer-based language model (PLM), such as BERT to embed these features. PLMs, even without any task-specific fine-tuning, provide rich, expressive latent representations for text. Therefore, we feed in the LLM-extracted textual features into a BERT (specifically, bert-base-uncased[1]) model and obtain the last hidden layer embedding of the `[CLS]` token, and we denote this as $h^i_{ft[CLS]}$.

Embedding Fusion & Classification From the previous two components, for each input text x_i, we have two embeddings: text embedding $h^i_{[CLS]}$ from the base hate speech detector, and feature embedding $h^i_{ft[CLS]}$ from the feature embedding BERT model. To combine these two, we simply concatenate these embeddings:

$$h^i_{combined} = h^i_{[CLS]} \oplus h^i_{ft[CLS]} \quad (1)$$

Note that while authors in (Jain et al., 2020) only use the extracted rationales in the subsequent detector model, we use a concatenated view in order to incorporate additional contextual features that may be very relevant to determining the hate or non-hate label (Ocampo et al., 2023). We then feed this combined embedding $h^i_{combined}$ into a feed-forward multi-layer perceptron with two fully connected layers and a ReLU activation (Agarap, 2018) in between, to project it onto a smaller dimension space. Following previous work (Pan et al., 2022; Bhattacharjee et al., 2023a), we do this in order to retain important features and avoid overfitting of the model during training. We denote this MLP as $f(\cdot)$. Finally we compute the batch-wise binary cross entropy loss using the ground truth label y_i for each input text x_i:

$$\begin{aligned} loss_{CE} = -\frac{1}{n}\sum_{i}^{n}[\log p(y_i|f(h^i_{combined}))+ \\ (1-y_i)\log(1-p(y_i|f(h^i_{combined}))] \end{aligned} \quad (2)$$

where n is the batch size. Since we are using the BERT feature embedding model just to encode the textual features z, we keep this model frozen and train the remainder of the framework with this simple loss.

3 Methodology and Experimental Settings

In this section, we discuss our methodology in detail including the datasets we included, the baseline models for hate speech detection along with the experimental settings.

Dataset	# of Posts	# of Hateful Posts	Hate %
GAB	14,240	11,920	83.7
Reddit	37,164	10,562	28.4
Twitter	10,457	3,933	37.6
YouTube	5,052	1,699	33.6
Implicit HS	20,391	7,100	34.8

Table 1: Dataset statistics for explicit and implicit hate speech datasets comprising data from different social media platforms.

3.1 Datasets

In order to evaluate **SHIELD**, we use both explicit and implicit hate speech datasets. For explicit hate, we include publicly available benchmark datasets from the following social media platforms: {GAB, Twitter, YouTube, and Reddit}. All these datasets are in the English language. **GAB** (Mathew et al., 2021) is a collection of annotated posts from the GAB website. It consists of binary labels indicating whether a post is hateful or not. **Reddit** (Kennedy et al., 2020) is a collection of posts indicating whether it is hateful or not. **Twitter** (Mathew et al., 2021) contains instances of hate speech gathered from tweets on the Twitter platform. Finally, **YouTube** (Salminen et al., 2018) is a collection of hateful expressions and comments posted on the YouTube platform. We further pre-process these according to the method followed in (Sheth et al., 2023a), in order to get cleaned binary labels. A summary of the datasets and the distribution of hateful posts and non-hateful posts can be found in Table 1.

We also include implicit hate speech in our evaluation: while subtle forms of abuse may not be perceived as overtly harmful initially, they nonetheless perpetuate similar degrees of damage over time owing to their covert nature. Therefore, the detection of implicit hate speech becomes even more important. For this reason, we evaluate our proposed model on the **Implicit Hate Speech Corpus** (ElSherief et al., 2018). This dataset encompasses posts compiled from Twitter, annotated as either explicit hate, implicit hate, or non-hate speech. We exclusively utilize implicit hate and non-hate for our binary classification task.

[1] https://huggingface.co/google-bert/bert-base-uncased

3.2 Baselines

We compare our proposed **SHIELD** framework to a variety of different baselines in order to understand the impact of the augmentation with rationales. We use the following well-known baseline hate speech detection models:

HateBERT: This is also the base model used in our framework. HateBERT (Caselli et al., 2020) uses over 1.5 million Reddit messages from suspended communities known for encouraging hate speech to fine-tune the BERT-base model. We further fine-tune HateBERT on each dataset and report the performance.

HateXplain: Similarly, we fine-tune the HateXplain (Mathew et al., 2021) model on each of our datasets and report the performance. HateXplain model is trained on hateful posts along with the target community, the rationales, and the portion of the post on which human annotators' labelling decision is based.

PEACE: We further extend our comparison on PEACE (Sheth et al., 2023b) framework which uses Sentiment and Aggression Cues to detect the overall sentiment of the text.

CATCH: Furthermore, we compare our model with CATCH (Sheth et al., 2023a) framework which disentangles the input representations into invariant and platform-dependent features.

ChatGPT-1shot: Apart from these hate speech specific detection models, we also compare our framework with an off-the-shelf **GPT-3.5** model, to understand how well the LLM performs on the same datasets. We do this in a one-shot manner, i.e., by proving the task instruction along with an example input and ground truth label.

3.3 Experimental Settings

To implement our proposed **SHIELD** framework, we use PyTorch and the Huggingface Transformers library. As shown in Figure 1, our first component uses an off-the-shelf LLM to extract the features and rationales. Here, we use OpenAI's GPT-3.5 (specifically, GPT-3.5-turbo-0613)[2], since it has been experimented on a variety of NLP tasks with huge success (Guo et al., 2024). We access this model via the OpenAI API. For feature/rationale extraction and generation, we set the temperature to 0.1 and top_p to 1. For the Feature Embedding Model we use a pre-trained, frozen BERT (bert-base-uncased) and for the Hate Speech Detector we use a pre-trained HateBERT[3] model. We use AdamW optimizer (Kingma and Ba, 2014) with a learning rate of 2×10^{-5}. Model training was performed on two machines: one with an NVIDIA GP102 [TITAN Xp] GPU with 12 GB VRAM, and another with an NVIDIA A100 GPU with 40GB RAM. For all detection experiments, we use accuracy as the evaluation metric.

4 Results and Discussion

In this section we describe our experiments and elaborate on the experimental results. To explore the feasibility and effectiveness of our proposed **SHIELD** framework, we aim to answer the following research questions:

- **RQ1:** How well does ChatGPT perform on our set of hate speech detection datasets?
- **RQ2:** Can we leverage recent state-of-the-art LLMs to extract features in the form of rationales, and do these rationales align with human judgement?
- **RQ3:** Can **SHIELD** effectively retain/improve performance of the hate speech detector while facilitating faithful interpretability?

4.1 Performance of ChatGPT on the hate speech detection task

Several recent works test whether Large Language Models have the potential to reproduce human annotated ground truth labels in social computing tasks (Zhu et al., 2023). However, even after extensive pre-training on a large corpus of datasets, where LLMs are expected to perform well in this task, this is not the case. To further evaluate this beyond what other recent works have shown, we carefully craft a one-shot prompt and prompt ChatGPT to classify the input text, given a labeled example in the prompt. The outcome of this prompt is a single label representing hateful text as label “1" and non-hateful text as label “0" as shown in Table 2.

We perform this classification using ChatGPT for all 5 datasets and compute the accuracy. We compare the results of this one-shot classification task with the baseline models (as described in Section 3.2) and show the results in Table 3. We see a stark difference in the performance of the baseline

[2] or otherwise commonly referred to as ‘ChatGPT’

[3] https://huggingface.co/GroNLP/hateBERT

Text	"always thought it astounding no policing agency dares do this to the armed muslim compounds all over the country"
Prompt	*You are a hate speech detection bot. Given a text, respond with the classification label towards given text with either hateful labeled as 1 or non-hateful labeled as 0. Note: It is essential to give classification for all the texts.* *Text:* "always thought it astounding no policing agency dares do this to the armed muslim compounds all over the country"
ChatGPT response	1

Table 2: Examples of input text, prompt and ChatGPT's response for a data sample from the Twitter dataset.

models vs. ChatGPT-1shot classification accuracies. While performance on the GAB dataset is satisfactory, ChatGPT struggles with the other 4 datasets with ~58-65% accuracy. Similar observations have been reported in other recent work that have investigated the off-the-shelf performance of LLMs in hate speech detection (Li et al., 2023; Zhu et al., 2023).

While this shows ChatGPT and possibly other LLMs struggle at hate speech detection when used as a detector directly, these models have also been shown to have impressive textual understanding capabilities. Perhaps, simply using these models to extract features or rationales, instead of performing the entire detection task, might be beneficial. We evaluate this in the following subsection.

4.2 Goodness of ChatGPT extracted features or rationales

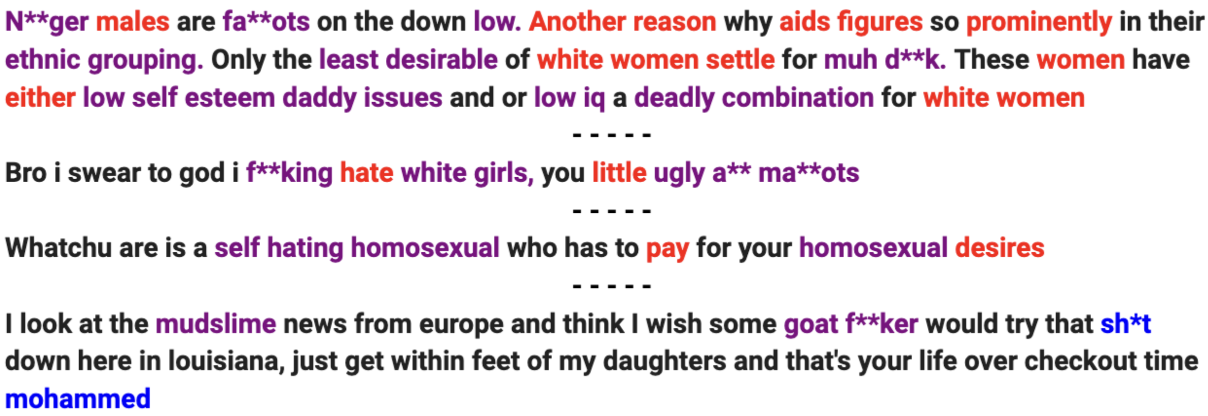
We are interested to evaluate the textual and contextual understanding capabilities of ChatGPT in order to extract features in the form of rationales from the input text that are meaningful to the task of hate speech detection. Following a similar construction as in (Jain et al., 2020), we use the LLM (i.e., GPT-3.5) as the *extractor* model, which unlike the extractor model in (Jain et al., 2020), does not require any additional task-specific finetuning. This is possible due to the instruction-following capabilities of recent LLMs. We carefully craft a prompt (as shown in Table 4) to extract *cuss words*, *derogatory language* and *rationales* from the input text that serve as interpretable features that can be used in the subsequent *predictor* model (HateBERT) in order to have a faithfully interpretable hate speech detector. In order to evaluate the goodness of the extracted features or rationales, we compare ChatGPT-extracted rationales with human-annotated ground truth rationales. We use the annotated rationale spans in the HateXplain (Mathew et al., 2021) dataset. After some standard pre-processing such as removing stop words, we compute the similarity between the ChatGPT extracted rationales for the input text from HateXplain dataset and the human-annotated rationales and report these scores in Table 5. We compute similarity metrics in both the token space (Jaccard and Overlap similarity) and in the latent space (Cosine and Semantic similarity with Universal Sentence Encoder embeddings (Cer et al., 2018)) We see significant overlap and a high semantic similarity between the LLM and human rationales.

We present some examples from all 5 datasets in Table 4: the input text with a 'hateful' label and the ChatGPT-extracted features. The three category of features are shown in different colors: rationales, derogatory language and cuss words. We see that the LLM is successfully able to identify the words and spans quite well.

We also present some examples in Figure 2 to qualitatively discern the overlap between the human-annotated rationales and the LLM-extracted ones. Text in red are rationales annotated by human annotators, text in blue are rationales or words identified by the LLM and text in purple are the spans where both the LLM and human annotations overlap. From these examples, we see that there is overall a high degree of overlap, and the LLM is able to capture semantically relevant portions of the text. Interestingly, we also see that while human annotators often annotate words or spans with lesser relevance to the task, the LLM extracted rationales do not contain these spans (such as 'aids figures' and 'prominently' in the first example in Figure 2). Using LLM-extracted rationales for training might be even more useful in such cases since some of the noisy signals in the data can be avoided.

	HateBERT	HateXplain	PEACE	CATCH	ChatGPT-1shot	SHIELD (ours)
GAB	96	**97**	91	82	85.39	96.3
YouTube	71	72	72	**79**	58.34	70
REDDIT	94	93	93	86	65.05	**94.5**
Twitter	56	60	31	**78**	60.09	64
Implicit HS	**78**	76	64	–	65.68	**78**

Table 3: Evaluation results (test set accuracy) for our **SHIELD** framework vs. the baseline models. Implicit HS refers to the Implicit Hate Speech Corpus. Values in **bold** denote the best performance, and underlined values denotes the second-best performance.



N**ger males are fa**ots on the down low. Another reason why aids figures so prominently in their ethnic grouping. Only the least desirable of white women settle for muh d**k. These women have either low self esteem daddy issues and or low iq a deadly combination for white women

- - - - -

Bro i swear to god i f**king hate white girls, you little ugly a** ma**ots

- - - - -

Whatchu are is a self hating homosexual who has to pay for your homosexual desires

- - - - -

I look at the mudslime news from europe and think I wish some goat f**ker would try that sh*t down here in louisiana, just get within feet of my daughters and that's your life over checkout time mohammed

Figure 2: Examples with both LLM-annotated and human-annotated rationales. Overlap is in purple.

Prompt: *You are a content moderation bot. Identify the list of rationales, list of derogatory language, list of cuss words that promote a hateful sentiment and respond with non-hateful if there are none. Note: The output should be in a json format. "Text"*

Dataset	Text	ChatGPT-extracted features and rationales
GAB	does this mean they arrested the mayor of london have cordoned off every muzrat ghetto in the uk every mosque closed down if not its business as usual	arrested the mayor of london, cordoned off every muzrat ghetto in the uk, every mosque closed down; muzrat, ghetto
REDDIT	"Shut up! 50 years ago we'd have you upside down with a f**king fork up your a**! You can talk, you can talk, you can talk! You're brave now motherf**ker! Throw his a** out, he's a n**ger! HE'S A N**GER! HE'S A N**GER! A N**GER! LOOK, THERE'S A N**GER!"	n**ger; f**king, motherf**ker
Twitter	"show me the black muslims bc im calling bs on this and you should too"	black muslims; black, muslims
YouTube	"Dirty f**king israeli zionist jews"	israeli, zionist, jews; f**king
Implicit HS	"i'm so glad #ferguson happened . you woke the world up to how much black people hate white people ."	ferguson, black people hate white people; black people, white people

Table 4: Examples from different datasets along with the LLM-extracted features and rationales. Rationales are in blue, derogatory language is in red, cuss words are in teal.

Similarity Metric	Similarity Coefficients (%)
Jaccard Similarity	60.39
Overlap Similarity	99.17
Cosine Similarity	74.51
Semantic Similarity (via USE)	56.09

Table 5: Similarity between HateXplain human explanations and LLM-extracted features/rationales.

4.3 Hate speech detector performance after training with extracted rationales

In this experiment, we try to train a hate speech detector with the extracted rationales additionally incorporated into the input text, to facilitate faithfully interpretable classifications. For this we use a HateBERT model as the base hate speech detector model and report results in Table 3, along with results from other baselines. We see that our **SHIELD** framework performs at par with a simple HateBERT fine-tuned on the same dataset, i.e., at par with the base model. This performance retention is encouraging, since models are otherwise known to trade-off accuracy for interpretability (Dziugaite et al., 2020; Bertsimas et al., 2019). Interestingly, in the Twitter dataset, we also see a significant 12.5% performance jump by our **SHIELD** model as compared to the fine-tuned HateBERT model. This potentially might be due to noise in the Twitter dataset: the extracted rationales may provide more discriminative training signals thus allowing the detector to train on robust features instead of noisy ones, although more analysis is required to verify this claim.

For some additional analysis on the effect of the framework components, we modify the choice of the base pre-trained language models in the two model components: the hate speech detector, and the feature extractor. The specific variations we experiment with are: (1) the original **SHIELD** framework which has HateBERT as the hate speech detector (HSD) and bert-base-uncased as the feature embedding model (FE), (2) **SHIELD** with a pre-trained roberta-base as the HSD instead of HateBERT and (3) **SHIELD** with a pre-trained roberta-base as the FE instead of bert-base-uncased. We choose to perform this analysis with roberta instead of the two bert based models since RoBERTa (Liu et al., 2019) has been shown to sometimes have better performance than BERT (Devlin et al., 2018) on a variety of natural language understanding tasks (Tarunesh et al., 2021). We report the results of this analysis in Table 6. Overall, we see some variation in performance on the model choice for the HSD and FE components. While roberta-base as the FE component marginally helps to improve performance for only one dataset, i.e., GAB, roberta-base as the HSD instead of HateBERT achieves higher performance for three datasets. This is particularly interesting since, unlike HateBERT, the pre-trained roberta-base is not specifically trained on the hate speech task.

Overall, **SHIELD** shows promising results in leveraging LLM-extracted rationales into augmenting a base hate speech detector, to facilitate faithful interpretability, while maintaining detection performance.

5 Related Work

5.1 Hate Speech Detection

There are two primary methods for approaching the detection of hate speech. Leveraging new or supplementary data is the first strategy. This involves making advantage of user attributes (del Valle-Cano et al., 2023), dataset annotator features (Yin et al., 2022), or comprehending the ramifications of hateful posts (Kim et al., 2022). One study, for instance, used the consequences of hateful posts to train a model on contrastive pairs that represent hate content in order to detect implicit hate speech (Kim et al., 2022). An additional study (Yin et al., 2022) brought to light the challenge of reaching agreement among annotators on subjective issues such as recognizing hate speech, and it recommended that definitive labels and annotator traits be included in training to improve the efficacy of detection. In a different study (del Valle-Cano et al., 2023), data from users' social situations and characteristics were analyzed to predict user satisfaction. But the problem with these strategies is that they could be challenging as access to auxiliary information across different platforms is seldom available.

The second tactic makes use of language models like BERT, which have been trained on large text datasets and are renowned for their capacity for generalization. The efficacy of these algorithms can be increased by fine-tuning them using particular hate speech datasets (Caselli et al., 2020; Mathew et al., 2021). One such example is HateBERT (Caselli et al., 2020), a model that was refined using over 1.6 million hostile remarks from Reddit and based on a BERT model. In a similar vein, HateXplain (Mathew et al., 2021)

	GAB	YouTube	REDDIT	Twitter	Implicit HS
SHIELD (roberta-base HSD)	87.53	**72.2**	84.8	**67.03**	**78.36**
SHIELD (roberta-base FE)	**96.42**	69.27	94.21	56.22	77.52
SHIELD	96.3	70	**94.5**	64	78

Table 6: Analysis of HSD and FE model choices in the **SHIELD** framework. HSD: hate speech detector, FE: feature embeddeing model. The original **SHIELD** framework has HateBERT as the hate speech detector and bert-base-uncased as the feature embedding model. Numbers in **bold** denote best performaning model variant for each dataset.

is another model created to recognize and interpret hate speech. Other strategies include concentrating on lexical indications (Schmidt and Wiegand, 2017) such as POS tags used (Markov et al., 2021), facial expressions, content-related portions of speech, or important phrases that communicate hate (ElSherief et al., 2018). In order to improve language model representations, one study manually determined that sentiment and hostility are causal cues (Sheth et al., 2023b). Another study leveraged a causal graph to disentangle the input representations into platform specific (hate-target related features) and platform invariant features to enhance generalization capabilities for hate speech detection (Sheth et al., 2023a). Although effective, this method also requires auxiliary data (such as hate target labels) which are seldom available across various platforms.

5.2 LLMs as Experts or Feature Extractors

Recent advancements in LLM research have demonstrated improved performance across not only many natural language tasks (Min et al., 2023), but also more challenging domains such as writing and debugging code, performing mathematical reasoning (Bubeck et al., 2023), etc. This has motivated a line of research where the community has been trying to evaluate how well these LLMs can perform different tasks. LLMs have shown promise in the task of data annotation (He et al., 2023; Bansal and Sharma, 2023), information extraction (Dunn et al., 2022), text classification (Kocoń et al., 2023; Bhattacharjee and Liu, 2024), and even reasoning (Ho et al., 2022). Given the ease with which these LLMs can be queried, these models often serve as faulty experts or pseudo oracles in many tasks. Past exploration has investigated whether language models can be used as factual knowledge bases (Petroni et al., 2019). A recent work has explored the possibility of using LLMs in the hate speech detection task (Kumarage et al., 2024). Similar to our approach, authors in (Hasanain et al., 2023) have tried to perform propaganda span annotation using language models. However, our approach focuses on leveraging the extracted spans, words and rationales to augment a detector model to enable interpretability in an otherwise black-box model.

6 Conclusion and Future Work

In this work, we explore the problem of hate speech detection on social media and propose a method to train interpretable classifiers using rationales extracted by large language models. Given the unsatisfactory performance of LLMs as off-the-shelf detectors for hate speech, we instead intend to leverage the textual understanding and instruction-following capabilities of LLMs such as ChatGPT to extract words and rationales from the text that are associated with the hate speech label. We propose a framework **SHIELD**, that uses these LLM-extracted rationales to augment the training of a base hate speech detector to facilitate it to be faithfully interpretable. We verify that the LLM-extracted rationales align with human judgement. We train and evaluate our framework on multiple benchmark datasets comprising both implicit and explicit hate speech from a variety of online social media platforms, and demonstrate how our **SHIELD** framework is able to maintain performance similar to the base model in spite of an expected accuracy-interpretability trade-off. Therefore, we have a faithfully interpretable hate speech detector that simply relies on LLM-extracted rationales instead of human-annotated.

While our work follows that of (Jain et al., 2020) and we establish faithfulness by construction, future work could explore better ways to evaluate the faithfulness of the resulting detector. In this work, we verified the goodness of the extracted rationales by comparing it with the ground truth for one dataset. Future work can investigate better automated ways to evaluate and verify the quality of the LLM-extracted rationales. Furthermore, an interesting and responsible direction forward would be the development of hybrid approaches that leverage LLMs for extracting rationales at scale and then employing human experts to verify the validity and quality of these rationales. This would also alleviate some of the concerns surrounding LLM hallucinations and biases in the LLM being propagated into the rationale extraction step.

7 Limitations

While our **SHIELD** framework shows promise in leveraging large language models to create interpretable hate speech detectors, several limitations need to be addressed. A inherent trade-off exists between the interpretability gained through LLM-extracted rationales and the accuracy of the resulting model, requiring further work to optimize this balance. In certain cases, the LLM may fail to identify coherent rationales, leading to incomplete or inaccurate explanations for the model's predictions. The choice of the LLM itself is also crucial, as powerful proprietary models like ChatGPT may not be accessible to all researchers, while open-source alternatives could potentially yield suboptimal performance. Our work currently uses ChatGPT for rationale extraction, but exploring the capabilities of different LLMs, including multilingual and domain-specific models, could provide valuable insights. Additionally, our framework may need adaptation to handle instances where the LLM cannot provide clear rationales, either through ensemble methods or by incorporating human feedback mechanisms to refine the extracted rationales.

8 Ethical Considerations

8.1 Acknowledgment of the sensitivity and potential harm of hate speech

We acknowledge that hate speech is a sensitive and potentially harmful topic that can perpetuate discrimination, marginalization, and violence against individuals or groups based on their race, ethnicity, religion, gender, sexual orientation, or other protected characteristics. We recognize the importance of addressing hate speech responsibly and with great care, as it can have severe psychological, emotional, and social consequences for those targeted. However, our work strives to better interpret and mitigate the use of hateful speech promptly by employing LLMs in an out-of-the-box manner leveraging their context-understanding capabilities in hate speech detection task.

8.2 Commitment to responsible use and mitigation of potential misuse

Our research focuses on leveraging the contextual understanding capabilities of large language models (LLMs) to automate the detection of hateful content, such as derogatory language, cuss words, and profanities, in the form of rationales across social media platforms. This aims to enable early-stage identification and mitigation of hate speech. We acknowledge the severity of the hateful examples used, which may potentially promote racial superiority, incite racial discrimination, or encourage violence against certain racial or ethnic groups – actions that are considered punishable offenses by law. After a thorough evaluation, we have concluded that the benefits of using real-world practical examples to enhance the clarity and understanding of our research outweigh any potential risks or drawbacks associated with their inclusion.

8.3 Ethical guidelines and principles followed

In conducting our research, we adhere to established ethical guidelines and principles, such as those outlined by professional organizations and academic institutions. We have utilized publicly available datasets that are appropriately cited in our paper. We also strive to maintain transparency by clearly documenting our methods, data sources, and limitations.

Acknowledgements

This work is supported by the DARPA SemaFor project (HR001120C0123), and the Office of Naval Research (N00014-21-1-4002). The views, opinions and/or findings expressed are those of the authors.

References

Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Parikshit Bansal and Amit Sharma. 2023. Large language models as annotators: Enhancing generalization of nlp models at minimal cost. *arXiv preprint arXiv:2306.15766*.

Dimitris Bertsimas, Arthur Delarue, Patrick Jaillet, and Sebastien Martin. 2019. The price of interpretability. *arXiv preprint arXiv:1907.03419*.

Amrita Bhattacharjee, Tharindu Kumarage, Raha Moraffah, and Huan Liu. 2023a. Conda: Contrastive domain adaptation for ai-generated text detection. *arXiv preprint arXiv:2309.03992*.

Amrita Bhattacharjee and Huan Liu. 2024. Fighting fire with fire: can chatgpt detect ai-generated text? *ACM SIGKDD Explorations Newsletter*, 25(2):14–21.

Amrita Bhattacharjee, Raha Moraffah, Joshua Garland, and Huan Liu. 2023b. Llms as counterfactual explanation modules: Can chatgpt explain black-box text classifiers? *arXiv preprint arXiv:2309.13340*.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2020. Hatebert: Retraining bert for abusive language detection in english. *arXiv preprint arXiv:2010.12472*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Gloria del Valle-Cano, Lara Quijano-Sánchez, Federico Liberatore, and Jesús Gómez. 2023. Socialhaterbert: A dichotomous approach for automatically detecting hate speech on twitter through textual analysis and user profiles. *Expert Systems with Applications*, 216:119446.

Fabio Del Vigna12, Andrea Cimino23, Felice Dell'Orletta, Marinella Petrocchi, and Maurizio Tesconi. 2017. Hate me, hate me not: Hate speech detection on facebook. In *Proceedings of the first Italian conference on cybersecurity (ITASEC17)*, pages 86–95.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alexander Dunn, John Dagdelen, Nicholas Walker, Sanghoon Lee, Andrew S Rosen, Gerbrand Ceder, Kristin Persson, and Anubhav Jain. 2022. Structured information extraction from complex scientific text with fine-tuned large language models. *arXiv preprint arXiv:2212.05238*.

Gintare Karolina Dziugaite, Shai Ben-David, and Daniel M Roy. 2020. Enforcing interpretability and its statistical impacts: Trade-offs between accuracy and interpretability. *arXiv preprint arXiv:2010.13764*.

Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. 2018. Hate lingo: A target-based linguistic analysis of hate speech in social media. In *Proceedings of the International AAAI Conference on Web and Social MediaProceedings of the International AAAI Conference on Web and Social Media*, volume 12.

Mary G Findling, Robert J Blendon, John Benson, and Howard Koh. 2022. Covid-19 has driven racism and violence against asian americans: perspectives from 12 national polls. *Health Affairs Forefront*.

Keyan Guo, Alexander Hu, Jaden Mu, Ziheng Shi, Ziming Zhao, Nishant Vishwamitra, and Hongxin Hu. 2024. An investigation of large language models for real-world hate speech detection. *arXiv preprint arXiv:2401.03346*.

Sungil Han, Jordan R Riddell, and Alex R Piquero. 2023. Anti-asian american hate crimes spike during the early stages of the covid-19 pandemic. *Journal of interpersonal violence*, 38(3-4):3513–3533.

Stefan Harrer. 2023. Attention is not all you need: the complicated case of ethically using large language models in healthcare and medicine. *EBioMedicine*, 90.

Maram Hasanain, Fatema Ahmed, and Firoj Alam. 2023. Large language models for propaganda span annotation. *arXiv preprint arXiv:2311.09812*.

Xingwei He, Zhenghao Lin, Yeyun Gong, Alex Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, Weizhu Chen, et al. 2023. Annollm: Making large language models to be better crowdsourced annotators. *arXiv preprint arXiv:2303.16854*.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2022. Large language models are reasoning teachers. *arXiv preprint arXiv:2212.10071*.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. 2020. Learning to faithfully rationalize by construction. *arXiv preprint arXiv:2005.00115*.

Chris J Kennedy, Geoff Bacon, Alexander Sahn, and Claudia von Vacano. 2020. Constructing interval variables via faceted rasch measurement and multitask deep learning: a hate speech application. *arXiv preprint arXiv:2009.10277*.

Youngwook Kim, Shinwoo Park, and Yo-Sub Han. 2022. Generalizable implicit hate speech detection using contrastive learning. In *Proceedings of the 29th International Conference on Computational LinguisticsProceedings of the 29th International Conference on Computational Linguistics*, pages 6667–6679.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Tharindu Kumarage, Amrita Bhattacharjee, and Joshua Garland. 2024. Harnessing artificial intelligence to combat online hate: Exploring the challenges and opportunities of large language models in hate speech detection. *arXiv preprint arXiv:2403.08035*.

Zachary Laub. 2019. Hate speech on social media: Global comparisons. *Council on foreign relations*, 7.

Lingyao Li, Lizhou Fan, Shubham Atreja, and Libby Hemphill. 2023. "hot" chatgpt: The promise of chatgpt in detecting and discriminating hateful, offensive, and toxic comments on social media. *ACM Transactions on the Web*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilia Markov, Nikola Ljubešić, Darja Fišer, and Walter Daelemans. 2021. Exploring stylometric and emotion-based features for multilingual cross-domain hate speech detection. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 149–159.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14867–14875.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2):1–40.

Nicolas Benjamin Ocampo, Ekaterina Sviridova, Elena Cabrio, and Serena Villata. 2023. An in-depth analysis of implicit and subtle hate speech messages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1997–2013. Association for Computational Linguistics.

Lin Pan, Chung-Wei Hang, Avirup Sil, and Saloni Potdar. 2022. Improved text classification via contrastive adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11130–11138.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Joni Salminen, Hind Almerekhi, Milica Milenković, Soon-gyo Jung, Jisun An, Haewoon Kwak, and Bernard Jansen. 2018. Anatomy of online hate: developing a taxonomy and machine learning models for identifying and classifying hate in online news media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10.

Paras Sheth, Tharindu Kumarage, Raha Moraffah, Aman Chadha, and Huan Liu. 2023a. Causality guided disentanglement for cross-platform hate speech detection. *arXiv preprint arXiv:2308.02080*.

Paras Sheth, Tharindu Kumarage, Raha Moraffah, Aman Chadha, and Huan Liu. 2023b. Peace: Cross-platform hate speech detection-a causality-guided framework. *arXiv preprint arXiv:2306.08804*.

Ishan Tarunesh, Somak Aditya, and Monojit Choudhury. 2021. Trusting roberta over bert: Insights from checklisting the natural language inference task. *arXiv preprint arXiv:2107.07229*.

Wenjie Yin, Vibhor Agarwal, Aiqi Jiang, Arkaitz Zubiaga, and Nishanth Sastry. 2022. Annobert: Effectively representing multiple annotators' label choices to improve hate speech detection. *arXiv preprint arXiv:2212.10405*.

Yiming Zhu, Peixian Zhang, Ehsan-Ul Haq, Pan Hui, and Gareth Tyson. 2023. Can chatgpt reproduce human-generated labels? a study of social computing tasks. *arXiv preprint arXiv:2304.10145*.