Causality Guided Disentanglement for Cross-Platform Hate Speech Detection

ABSTRACT

Despite their value in promoting open discourse, social media platforms are often exploited to spread harmful content. Current deep learning and natural language processing models used for detecting this harmful content rely on domain-specific terms affecting their ability to adapt to generalizable hate speech detection. This is because they tend to focus too narrowly on particular linguistic signals or the use of certain categories of words. Another significant challenge arises when platforms lack high-quality annotated data for training, leading to a need for cross-platform models that can adapt to different distribution shifts. Our research introduces a cross-platform hate speech detection model capable of being trained on one platform's data and generalizing to multiple unseen platforms. One way to achieve good generalizability across platforms is to disentangle the input representations into invariant and platform-dependent features. We also argue that learning causal relationships, which remain constant across diverse environments, can significantly aid in understanding invariant representations in hate speech. By disentangling input into platform-dependent features (useful for predicting hate targets) and platform-independent features (used to predict the presence of hate), we learn invariant representations resistant to distribution shifts. These features are then used to predict hate speech across unseen platforms. Our extensive experiments across four platforms highlight our model's enhanced efficacy compared to existing state-of-the-art methods in detecting generalized hate speech.

ACM Reference Format:

. 2023. Causality Guided Disentanglement for Cross-Platform Hate Speech Detection. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Warning: *this paper contains contents that may be upsetting.*

The worldwide adoption of social media platforms has led to a proliferating volume of social exchanges. The communications happening over these platforms can strongly influence public opinions. One form of communication influencing social beliefs and opinions is hate speech. As the name suggests, hate speech refers to those communications that encourage or support acts of violence, prejudice, hostility, or discrimination against an individual or a group based on social characteristics such as race, ethnicity, religion, etc. Hate speech content uses derogatory and insulting language to denigrate specific people or groups. The effects of hate speech extend beyond mental distress for its victims. Numerous studies have demonstrated the link between hate speech, real-world violence, and hate crimes [1, 44]. Thus, given the heinous effects and the aftermath of hate speech, it becomes imperative to detect such content on online social media platforms and prevent its spread for a respectful and sustainable online environment.

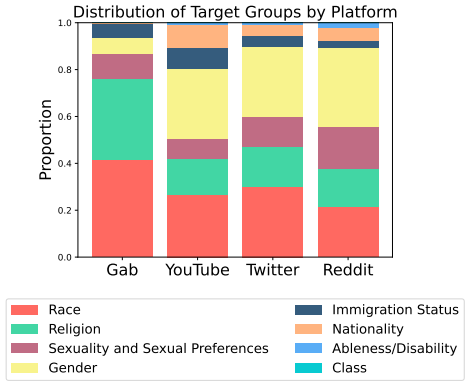


Figure 1: The distribution of the target of hate speech varies across different platforms, indicating that the target can be leveraged as a platform-dependent feature.

Due to the volume of online content, it is infeasible to detect hate speech manually. Thus, researchers leverage deep learning and natural language processing-based models to automate hate speech detection [16]. However, even automatic hate speech detection is hindered by several existing challenges. One major challenge is the *limited availability of labeled data* for some platforms. Over the recent years, there has been a frequent emergence of new social media platforms [18]. Not all these platforms have access to quality annotated data. As a result, to employ hate speech detection models for these platforms, one sensible approach is to leverage available quality training data from other platforms (source). However, if the model does not adjust for the biases present in the training data of the source platform, the model may end up failing to detect hate speech on the target platform [27]. Another critical challenge that hinders the performance of the existing models is the *over-reliance* on linguistic cues and certain category of words (e.g., profane words) to detect hate speech. A range of work [37, 47] show that the current state-of-the-art models overly rely on identity words (e.g., "jews") to predict hateful content. Moreover, some studies [7, 13] highlight how the existing models consider content with profane words to be mostly hateful. The replication of such biases can seriously impact the model's capabilities to perform well across platforms. Thus, there is a need for addressing these challenges to enable effective cross-platform generalization for hate speech detection.

A cross-platform hate speech detection model is trained on data from one source domain or platform and applied to unseen target domains. These cross-platform models seek to identify the underlying patterns in hate speech rather than flagging specific words or phrases often associated with the content on a platform. This approach is advantageous as it adapts to changes in how different

Conference'17, July 2017, Washington, DC, USA

© 2023 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

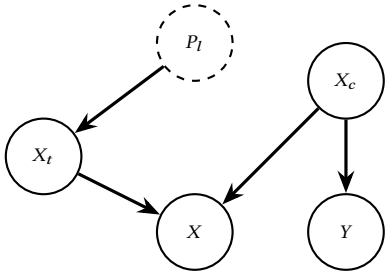


Figure 2: The causal graph representing the data generating mechanism for hate speech detection. X_c represents the causal representation useful for predicting whether the content is hateful or not, Y represents the hate label, X_t the target of hate speech, X the input, and P_l the latent platform variable influencing the target.

people use language, avoiding over-reliance on specific words [37] and aids social media platforms that lack quality labeled data.

Recent works aim to build such generalizable models for detecting hate. For instance, the authors of [20] leverage auxiliary information and contrastive learning to determine whether a post is implicitly hateful. However, the auxiliary information may not be available for large datasets or different platforms, limiting the framework's applicability. Another line of work [29, 32] considered the POS tags and the emotions for training a generalizable hate speech model. Such methods tend to form a spurious correlation with certain tags (e.g., adverbs) or certain kinds of words (e.g., profane words), thus degrading its generalization power. The authors of [41] identified two quantifiable invariant cues, namely, aggression and overall sentiment present in text to guide representations to be generalizable for hate speech detection. This approach identifies cues manually, however, more cues might exist that can help guide the representations. Pretrained language models such as BERT have also been fine-tuned on large hate speech datasets to create specialized hate detection models, such as HateBERT [6] and HateXplain [31]. However, these models tend to replicate any biases present in the datasets they were fine-tuned on, which can limit their effectiveness when applied to new or different contexts.

In light of the intricate nature of hate speech detection and the challenges posed by existing state-of-the-art models that strive for cross-platform hate speech detection, we propose to explore ageneralizable hate speech detection model that overcomes these limitations. A prevalent and proven technique for constructing a generalizable model involves leveraging the concept of disentanglement. This paper is the first attempt to adapt the disentanglement technique to cross-platform hate speech detection. Disentanglement segregates the input representation into a platform-dependent component containing platform-specific information and an invariant component containing information shared across different platforms. Causality is recently explored to capture the invariance across different platforms [26, 42]. Causal features represent the causal relationships or mechanisms underlying a particular phenomenon. When inferred that a variable causes another variable, it is implied that a universal and consistent relationship exists. Identifying and separating such factors allows us to understand the true underlying structure of the data. Furthermore, causal relations are known to be generally invariant, thus can enhance the generalization capabilities of machine learning models [4]. Thus, a causality-guided disentanglement model demands identifying the platform-dependent features w.r.t. hate speech detection problem.

Investigating the recent empirical analysis [3] brings to attention that the distribution of the targets of hate speech varies across platforms, allowing the target of hate speech to be a potential candidate for platform-dependent features. To verify this finding, we considered hate speech datasets from four platforms, i.e., Gab, YouTube, Twitter, and Reddit. The distribution of the target categories across platforms can be seen in Figure 1. It is evident that a platform can influence the choice of the target of hate speech. This is aligned with previous findings where researchers showed platforms such as YouTube, and Reddit are more susceptible to hate speech based on people's gender [9, 11] and platforms such as Twitter and GAB are more susceptible to hate speech based on race [30, 43]. Thus, considering the target as a platform-dependent variable can facilitate the disentanglement process. To this end, we propose to formulate the data-generating mechanism of the hate speech problem using the causal graph as shown in Figure 2. The causal graph aligns with the intuition that the core properties of hate remain invariant w.r.t. the hate target; what changes with the target is the intensity of these properties. For instance, a person may be more hurtful towards someone's race and less intense towards religion. Hence, segregating the platform-dependent component (target) from text representations can aid in learning generalizable representations.

Based on the aforementioned observations, we propose a novel **CA**usality-aware disen**T**anglement framework for **C**ross-platform **H**ate speech detection, namely, **CATCH** that leverages the causal graph shown in Figure 2 to disentangle the input representations into two parts: (i) a target representation (indicating platform-dependent features) and (ii) causal representations (indicating the platform invariant features), to facilitate learning generalizable hate speech representations across different platforms. We summarize our main contributions as follows:

Our Contributions

- We propose a causal graph showing the data-generating mechanism for the hate speech detection problem.
- We propose a novel causality-aware disentanglement framework, **CATCH**, that follows the causal graph and disentangles the overall representations of the content into two parts, one for platform-dependent and one for platform invariant features, to learn generalizable representations across different platforms.
- Experimental results on four platforms demonstrate that **CATCH** outperforms the state-of-the-art baselines.

2 RELATED WORK

Several studies have explored generalizable hate speech detection from various angles. Some have proposed new techniques to enhance generalization [8, 20, 46], while others have conducted thorough analyses to determine factors such as model choice, intra-dataset performance, and classification type that impact generalization [2, 3, 5, 14, 28]. In this section, we cover methods and frameworks related to these focuses.

2.1 Empirical Analysis of Hate Speech Generalization

The authors of [14] trained four models on nine English datasets and analyzed their generalization capabilities. They concluded that the ease of generalizing varies by model and category, with labels such as 'toxic' or 'offensive' being more manageable than 'hate speech.' They also showed that the target of hate speech can influence generalization. Similarly, the authors of [3] revealed how generalizability degrades across different hate speech topics, demonstrating that current state-of-the-art (SOTA) models trained on specific topics struggle to generalize to unseen ones.

Some studies specifically analyze platform-based hate speech, such as on Twitter [28] and YouTube [5]. The authors of [28] assess various methods on Twitter, revealing how hate speech is distributed among different categories. The authors of [5] examine hate against Afro-descendant, Roma, and LGBTQ+ communities within YouTube comments. Another study [2] emphasizes how biases in the data creation process affect model generalization. Drawing from these insights, we conclude that hate target distribution varies across platforms. Therefore, by isolating the target-dependent aspect of input representation, we can use the remaining component to create generalizable hate representations.

2.2 Methods to Enhance Hate Speech Generalization

Two primary techniques can be used to generalize hate speech detection in general. First, some models make use of extra or auxiliary data, such as user attributes [8], dataset annotator features [46], or implications of hateful posts [20]. By exploiting the implications of hateful posts to train contrastive pairs for a universal representation of hate content, one study [20], for instance, proposed a model that could generally recognize implicit hate speech. Another study [46] made the case that it is challenging for annotators to agree on subjective tasks like the identification of hate speech, and it thus suggested using the annotator's traits and the ground truth label during training to improve hate speech detection. In contrast, another study [8] trained a model for predicting user satisfaction using data from their social context and their profiles. However, the issue with these models is that the auxiliary information they rely on might not always be easily obtainable, especially in cross-platform scenarios.

Second, some techniques use language models like BERT, which have generalization abilities because they were trained on vast text corpora. By adjusting these models using particular hate speech datasets, they can be further enhanced [6, 31]. An illustration of this is HateBERT [6], a cutting-edge model for hate speech identification developed by optimizing a BERT model on around 1.6 million hostile messages from Reddit. There are also models like HateXplain [31], developed to identify explicable hate speech. In addition to these strategies, other techniques take lexical signals into account [40], such as the language employed, emotion words, POS tags in the material [29], or keyphrases that target hatred [10]. Another work [41] manually identified two causal cues, aggression and sentiment, to generalize the language model representations.

Although these techniques have improved the ability to identify hate speech, they have their share of challenges. For instance, fine-tuning language models may not be feasible as they require large labeled datasets specific to the task. Additionally, since many social media posts contain grammatical errors like misspelled words, depending on lexical features may be ineffective. Furthermore, manually identifying cues is an exhaustive task and might overlook some important aspects, such as cues like sentiment and aggression are applicable for more explicit settings compared to implicit scenarios. To address these challenges and improve the generalization capabilities, we aim to causally disentangle the representations to segregate platform-dependent features from the overall representation, leading the remainder to be invariant and easily generalizable.

3 PROPOSED METHOD

3.1 Preliminaries

Let D_{source} denote a source corpus composed of a series of textual inputs (tweets or posts) $X = \{x_1, x_2, ..., x_n\}$, a corresponding series of labels indicating whether the posts are hateful or not $Y = \{y_1, y_2, ..., y_n\}$, and a series of labels denoting the targeted group of the post $T = \{t_1, t_2, ..., t_n\}$. In the context of generalizable hate speech detection, our primary aim is to learn a mapping function $f : X \rightarrow Y$ capable of determining a generalizable representation from D_{source}. Such a representation enables the accurate prediction of the hate label across previously unseen target domains D_{target}.

Our method seeks to augment this generalization capability by adopting the disentanglement principle. This approach is inspired by recent research findings demonstrating a significant variation in the hate target across different platforms. Therefore, we opt to represent the hate speech problem through a causal graph illustrated in Figure 2. The proposed model, **CATCH**, comprises a VAE leveraged for disentanglement and a prediction module where the disentangled representations are leveraged for prediction. The cornerstone of our method lies in disentangling the input representations into two distinct components: (1) a component (X_t) that embodies information pertinent to the hate target, and (2) a component (X_c) that encapsulates the causal aspects crucial to determine whether the content is hateful or not. This disentanglement paves the way for leveraging the generalized hate representations learned from one platform (or domain), enabling us to apply these insights to previously unseen domains.

3.2 Disentangling Causal and Target Representations

Expressing hate involves two components: the target of hate and the content that renders a post hateful. Our research proposes a causal graph and hypothesis suggesting that hate's target and intensity vary across platforms. On the other hand, the fundamental properties of hate, both abstract and quantifiable, remain constant

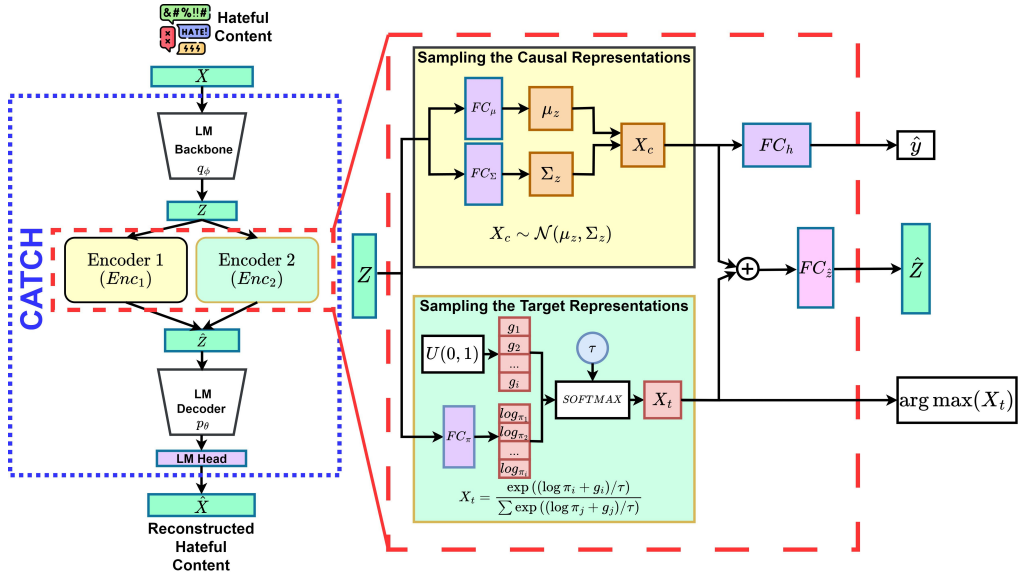


Figure 3: Overall architecture of CATCH. The hateful content X first passes through the LM Backbone to obtain the initial input representation Z. The representation Z then passes through two components - namely the continuous encoder (Enc_1) and the discrete encoder (Enc_2), where the continuous causal representations X_c are generated from $\mathcal{N}(\mu_z, \Sigma_z)$, and the discrete target representations X_t are generated from $\mathcal{F}(\pi, g)$. X_c is used to predict the hate label $\hat{y}$, and $\arg\max(X_t)$ represents the target label. X_c and X_t are concatenated to obtain the reconstructed embedding $\hat{Z}$ passed through the LM decoder to obtain the reconstructed input $\hat{X}$. FC represents a fully connected layer, $U(0,1)$ represents a uniform distribution and τ represents the temperature.

and can be considered causal factors. Our proposed **CATCH** employs the Variational AutoEncoder (VAE) framework to capture these causal representations. The model consists of an encoder, a disentanglement module, and a decoder.

The encoder q_ϕ represents a language model, such as Roberta, which takes an input text x and generates the embedding z as follows:

$$z = q_\phi\left(\gamma(x)\right), \tag{1}$$

where, $z \in \mathbb{R}^{S \times h_d}$ represents the language model representation, $\gamma(x)$ represents the tokenizing function of the language model, S indicates the sequence length, and h_d is the embedding dimension. We utilize the embedding of the start-of-sentence token ([CLS]) located at position 0, denoted as $z^{[CLS]} \in \mathbb{R}^{h_d}$, as the input representation for x.

Disentangling the Causal Component To disentangle z and obtain the causal counterpart X_c, we employ a VAE architecture. Disentangled representation refers to a factorized representation, where each latent variable corresponds to a single explanatory variable responsible for generating the data. VAEs are well-suited for achieving this task, as they impose a standard Gaussian prior distribution on the latent space and approximate the posterior using a parameterized neural network. Two feedforward neural networks, FC_μ and FC_Σ, are employed to map z to the Gaussian distribution parameters of hate. The latent representation $X_c \in \mathbb{R}^{h_{causal}}$ is sampled from the Gaussian distribution defined by the corresponding (μ_z, Σ_z) using the re-parameterization trick [21], where h_{causal} represents the hidden dimension size:

$$X_c = Enc_1(\mu_z, \Sigma_z) = \mu_z + \Sigma_z \odot \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{2}$$

Here, $\mu_z = FC_\mu(z)$, $\Sigma_z = FC_\Sigma(z)$, and $\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ represents the normal distribution, and $\mathbf{I}$ represents unit variance.

Disentangling the Target component Besides the causal component, z also includes platform-dependent features, among which is the target of hate. Recent studies reveal variations in hate speech across different platforms. For instance, YouTube comments predominantly manifest hate based on sexism [9], while Twitter is more affected by racism-based hate [43].

However, unlike the causally invariant hate features, the platform-dependent features related to the target are discrete. Hate speech targets can be categorized into eight broad classes: Race, Religion, Gender, Sexual Preference, Nationality, Immigration, Disability, and Class [3]. Consequently, the latent representation for the target is less likely to be Gaussian and more likely to be categorical or discrete in nature [12]. To effectively model the discrete latent variable, we employ VAEs leveraging the "Gumbel-Softmax" trick [17]. This enables the reparameterization of discrete variables within the VAE framework, facilitating the use of backpropagation for optimization and learning a disentangled representation [17].

The process commences by using a feed-forward neural network FC_π to map z to the latent representation $z_\pi \in \mathbb{R}^{h_{disc}}$ for the discrete variable, where h_{disc} represents the hidden dimensions. z_π is obtained as $z_\pi = FC_\pi(z)$.

Let z_π represent the categorical variable with class probabilities $\{\pi_1, \pi_2, \cdots \pi_{h_{disc}}\}$. We then apply the logarithmic function to each probability and add Gumbel noise, sampled by taking two logs of a uniform distribution. Let g_i represent the i.i.d. samples drawn from

Gumbel (0,1) [17]. To ensure a continuous, differentiable approximation , and generate h_{disc}-dimensional sample vectors, we further leverage the Softmax function to obtain X_t:

$$X_t = Enc_2(\pi, g) = \frac{\exp\left(\left(\log\left(\pi_i\right) + g_i\right)/\tau\right)}{\sum_{j=1}^{h_{disc}} \exp\left(\left(\log\left(\pi_j\right) + g_j\right)/\tau\right)} \quad \text{for } i = 1, \ldots, h_{disc}. \tag{3}$$

where τ represents the temperature.

Reconstructing the Input from the Disentangled components To facilitate the training, we aim to reconstruct the input from the obtained components X_c and X_t. To do so, we first concatenate them together as $[X_c \mid X_t]$ where [|] is the concatenation operation. Then we pass it through another feed-forward neural network, namely, $FC_{\hat{z}}$. The obtained representation after concatenation is given by $\hat{z} = FC_{\hat{z}}([X_c||X_t])$. To recreate the input, we feed $\hat{z}$ through a Language Model (LM) decoder $p(x|\hat{z})$. We utilize BART-base decoder [45] as the LM-decoder, as it shares the vocabulary with RoBERTa [45] implementations and is proved powerful in many generative tasks. The obtained tokens are given as follows:

$$\hat{x} = LMHead(p_\theta(\hat{z})) \tag{4}$$

where $LMHead$ is a feed-forward layer to map the decoder (p_θ) embeddings into tokens. The reconstructed loss is computed between the input token ids and the reconstructed token ids and is formulated as follows:

$$\mathcal{L}_{recon}(\gamma(x), \hat{x}) = -\sum_{i=1}^{S} \gamma(x) \log(\hat{x}_i) \tag{5}$$

where $\mathcal{L}$ represents the cross entropy loss, and S is the sequence length. To ensure that both the disentangled latent spaces' posteriors are close to their prior distribution, we further enforce KL-divergence losses for both latent spaces. The Evidence Lower BOund (ELBO) for the disentanglement module is formulated as follows:

$$\mathcal{L}_{VAE} = \mathcal{L}_{recon} + \alpha_t * \mathrm{L}_{\mathbb{D}_{target}} + \alpha_c * \mathrm{L}_{\mathbb{D}_{causal}}, \tag{6}$$

where α_t represents the coefficient that controls the contribution of the KL loss for the target, and α_c represents the coefficient that controls the contribution of the causal KL loss. To facilitate learning from the target labels, we follow [21] and add a cross-entropy loss to the target's KL loss. $\mathrm{KL}_{\mathbb{D}_{target}}$ is given by,

$$\mathrm{L}_{\mathbb{D}_{target}} = -D_{\mathrm{KL}}\left(Enc_2(X_t \mid X) \| p(X_t)\right) + \alpha_{tc} * \mathcal{L}_{CE}(\arg\max(X_t), t) \tag{7}$$

where t is the ground truth target label, $\arg\max(X_t)$ is the predicted target label given by, D_{KL} represents the KL divergence, $\mathcal{L}_{CE}$ is the cross entropy loss and α_{tc} is the coefficient controlling the contribution of the target classification loss. Similarly, $\mathrm{KL}_{\mathbb{D}_{target}}$ is given by,

$$\mathrm{L}_{\mathbb{D}_{causal}} = -D_{\mathrm{KL}}\left(Enc_1(X_c \mid X) \| p(X_c)\right) \tag{8}$$

3.3 Model Training

For hate speech detection, the disentangled latent causal representation X_c is utilised to compute the classification probability:

$$\hat{y}_i = \mathrm{Softmax}(FC_h(X_c)) \tag{9}$$

where FC_h represents a fully connected layer for hate classification. Then we compute the overall loss $\mathcal{L}$ using standard cross-entropy:

$$\mathcal{L}_{hate} = -\frac{1}{N} \sum_{i=1}^{|D_{source}|} y_i \log \hat{y}_i \tag{10}$$

where y_i represents the true hate label, and $\hat{y}_i$ denotes the predicted hate labels. Finally, we combine all proposed modules and train in a multi-task learning manner:

$$\mathcal{L} = \mathcal{L}_{hate} + \mu_d \mathcal{L}_{VAE} \tag{11}$$

where the μ_d is a pre-defined weight coefficient.

4 EXPERIMENTS

In this section, we implement a set of experiments designed to validate the effectiveness of **CATCH** for learning generalizable representations for detecting hate speech with causality-aware disentanglement. We make use of multiple datasets procured from various online platforms to provide a comprehensive evaluation of our methodology. Further, we conduct an extensive analysis encompassing ablation tests, and interpretability analysis to dissect the inner workings of our model. Moreover, we conduct a case study comparing how **CATCH** performs in comparison with the large scale language models such as Falcon [34], and GPT-4 [35]. The objective of these experimental analyses is to answer the following set of pertinent research questions:

- **RQ.1** Can the disentanglement of the input representation into a causal component and a platform-dependent component (target) aid in learning invariant causal representations that can improve the generalizability of hate speech detection?
- **RQ.2** Are the learned disentangled representations invariant across the different platforms?
- **RQ.3** What is the contribution of **CATCH's** components in aiding with the generalizability of representations?

Datasets	No. of Posts	Hateful Posts	Hate %
GAB [31]	11,093	8,379	75.5
Reddit [19]	37,164	10,562	28.4
Twitter [31]	9,055	2,406	26.5
YouTube [39]	1,026	642	62.5

Table 2: Dataset statistics with corresponding platforms and percentage of hateful comments or posts.

4.1 Datasets and Evaluation Metrics

We perform binary classification of hate speech detection on widely used benchmark hate datasets. Since we aim to verify cross-platform generalization, we use data from four different platforms for cross-platform evaluation: GAB, Reddit, Twitter, and YouTube. All datasets are in the English language. GAB [31] is a collection of annotated posts from the GAB website. It consists of binary labels indicating whether a post is hateful or not. These instances are annotated with corresponding explanations, where crowd workers justify why the given post, or content is considered hateful.

Reddit [19] is a collection of posts indicating whether it is hateful or not. It contains ten ordinal labels (sentiment, (dis)respect, insult,

Source	Target	Models					
		Easy Mix	Hate Bert	Hate Xplain	POS+ EMO	PEA- -CE	CAT- -CH
GAB	GAB	0.70	**0.89**	0.87	0.77	0.76	0.82
	Reddit	0.62	0.66	0.66	0.56	0.69	**0.72**
	Twitter	0.64	0.63	0.65	0.44	0.64	**0.69**
	YouTube	0.62	0.60	0.62	0.50	0.64	**0.66**
Reddit	GAB	0.51	0.52	0.56	0.45	0.55	**0.58**
	Reddit	0.95	**0.98**	0.94	0.91	0.90	0.86
	Twitter	0.54	0.51	0.54	0.43	0.55	**0.60**
	YouTube	0.64	0.69	0.60	0.57	0.70	**0.76**
Twitter	GAB	0.62	0.63	0.62	0.56	0.65	**0.67**
	Reddit	0.64	0.62	0.62	0.48	0.66	**0.69**
	Twitter	0.67	**0.86**	0.83	0.68	0.63	0.78
	YouTube	0.65	0.59	0.63	0.53	0.64	**0.68**
YouTube	GAB	0.44	**0.62**	0.47	0.43	0.48	0.56
	Reddit	0.67	0.65	0.62	0.56	0.69	**0.72**
	Twitter	0.45	0.59	0.56	0.49	0.58	**0.64**
	YouTube	0.86	0.84	**0.88**	0.64	0.86	0.79

Table 1: Cross-platform and in-dataset evaluation results for the different baseline models compared against CATCH. Boldfaced values denote the best performance, and the underline denotes the second-best performance.

humiliation, inferior status, violence, dehumanization, genocide, attack/defense, hate speech), which are debiased and aggregated into a continuous hate speech severity score (hate speech score). We binarize this data so that any data with a hate speech score less than 0.5 is considered non-hateful. Twitter [31] contains instances of hate speech gathered from tweets on the Twitter platform. Similar to the Gab dataset, these instances are also paired with explanations written by crowd workers, aiming to explain the hatefulness present in the respective tweets. Finally, YouTube [39] is a collection of hateful expressions and comments posted on the YouTube platform. All these datasets contain the hate labels as well as the target labels. A summary of the datasets can be found in Table 2. We use macro F1-measure for validation.

4.2 Experimental Setting

Implementation Details We trained our framework **CATCH** using RoBERTa-base for the LM-encoder and BART-base for the LM-decoder with the Huggingface Transformers library. We optimized the model with cross-entropy loss and AdamW, using a learning rate of 0.0001, dropout rate of 0.2, and parameters α_t as 0.05, α_c as 0.05, α_{tc} as 0.001, and μ_d as 0.5. Training was conducted on an NVIDIA GeForce RTX 3090 GPU with 40 GB VRAM, with early-stopping.

Baselines We compare our **CATCH** framework against various state of the art baselines. These baseline methods were designed to enhance the generalization prowess for cross-platform hate speech detection. The details of the methods categories is shown below:

- **EasyMix** [38] leverages data augmentation to generate training samples that cover a wide distribution of hateful language.
- The **POS+EMO** baseline [29] uses linguistic cues like POS tags, stylometric features, and emotional cues.
- **HateBERT** [6] uses over 1.5 million Reddit messages from suspended communities known for encouraging hate speech to finetune the BERT-base model. We further finetune HateBERT on each source domain and report the performance.
- Utilizing hate speech detection datasets from Twitter and Gab, **HateXplain** [31] is improved with an emphasis on a three-class classification task (hate, offensive, or normal) with human-annotated justifications. We also fine-tune HateXplain on each source domain and then report its performance.
- **PEACE** [41] leverages two causal cues, namely, the sentiment and aggression, to make representations more generalizable.

4.3 Performance Comparisons (RQ.1)

In our study, using a variety of real-world datasets, we compare various baseline models with **CATCH** on four different platforms. Each dataset is divided into train and test sets to evaluate the generalizability of these models. Then, each model is trained using data obtained from a single platform and assessed against test sets obtained from all platforms. The performance of several test sets is compared using the macro-F1 measure as a benchmark in Table 1. The **Target Platforms** column lists the platforms used for model assessment, while the **Source Platform** column provides the platform used for model training. We note the performance of each model for each source dataset in cross-platform and within-platform situations. We discern the following observations regarding the cross-platform performance regards to RQ.1:

- Predominantly, the **CATCH** delivers the most efficient performance in a cross-platform scenario. On average, the Model enhances cross-domain efficiency by approximately 3% on GAB, 5% on Reddit, 3% on Twitter, and 3% on YouTube. We attribute this improvement to applying causality-aware disentanglement, which initially distinguishes the platform-dependent target from

the platform-invariant causal representations, utilizing only the causal representations for hate prediction. Moreover, these results further corroborate the hypothesis that the intensity of hate varies with target, while the essential hate attributes remain consistent.

- Among the comparative baselines, PEACE is the most effective in most cases, followed by HateBERT. These outcomes align with prior findings and underscore two critical aspects. Since PEACE strives to identify and employ invariant causal indicators (sentiment and aggression) to steer the representations, we can deduce that causality can result in superior generalization by promoting invariance [4]. Additionally, fine-tuning language models can enhance performance. HateBERT and HateXplain are fine-tuned BERT models on extensive corpora of hateful content, which we further fine-tune for each source platform. Notwithstanding its potency, we argue that PEACE does not supersede the Model because not all hateful statements inherently contain sentiment and aggression [36]. Furthermore, while PEACE manually identifies only two hate attributes, others like context and socio-political climate are also significant [15].
- The EasyMix baseline encounters challenges in generalizing in certain cases. It employs data augmentation and frames the hate speech task as an entailment issue. Nevertheless, we speculate that its difficulty in outperforming other models arises from its reliance on varied combinations of the samples in the source domain, such as mixing non-hateful samples with other non-hateful samples. Such an approach might enhance the dependence on incidental correlations, thus benefiting in-domain performance but negatively impacting out-of-domain performances.
- The baseline based on linguistic features (POS + EMO) doesn't generalize well to these datasets. We postulate that this is due to the highly unstructured and grammatically incorrect nature of the posts in these datasets. Even following pre-processing, the inferred POS tags and emotion words may not accurately reflect the hateful content. Consequently, the reliance on these features impairs the generalization performance.

4.4 Are the Disentangled Causal Representations Invariant? (RQ.2)

One of the fundamental characteristics of the **CATCH** is its capacity to disentangle input representations into causal and platform-dependent (target) elements. The causal representations are expected to exhibit invariance as they are employed to predict the hate label. To validate the presence of this invariance, we carry out an additional experiment. In this, we train the **CATCH** on GAB, which serves as our source platform. We then examine the t-SNE plot of the causal representations acquired from various platforms by randomly sampling 1,000 instances of hateful and non-hateful posts. The extent of overlap between these representations indicates the model's ability to extract invariant features, which are crucial for generalization [26]. For an equitable comparison, we extend this experiment to HateBERT, PEACE, and HateXplain, our top three baseline models, and the results are illustrated in Figure 4.

As can be discerned from the figure, the causal representations generated by the **CATCH** display a high degree of overlap compared to those produced by HateBERT, HateXplain, and PEACE.

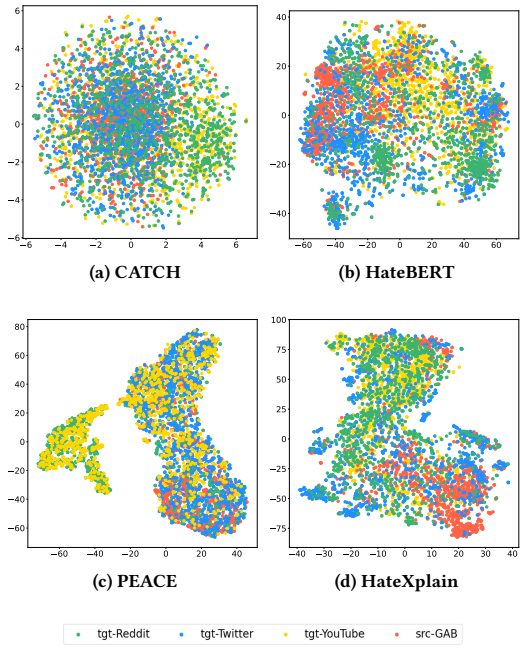


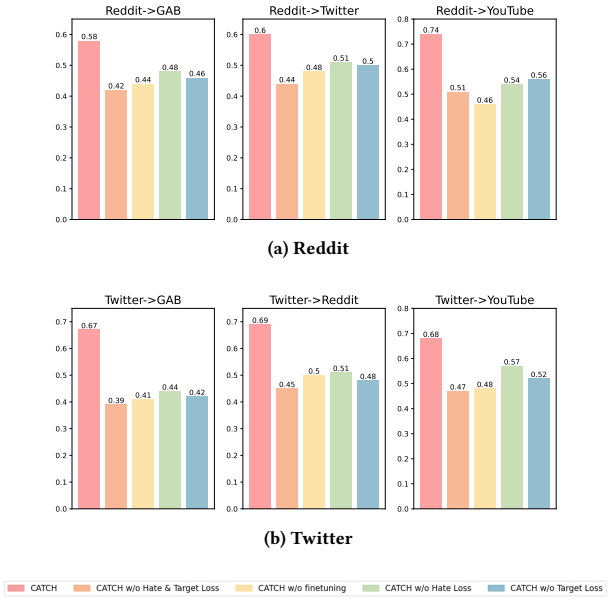
Figure 4: Measuring whether the disentangled causal representations are invariant across the different platforms. "src" denotes the source and "tgt" denotes the target platforms.

This suggests that the **CATCH** is adept at learning invariant features universally shared across the domains. Furthermore, PEACE's representations exhibit more overlap than those of HateBERT and HateXplain. This is consistent with the fact that PEACE identifies two intrinsic causal cues common to various platforms. Despite this, PEACE does not outperform the **CATCH**. This could be because while PEACE manually identifies two cues, there might be other unidentified cues (e.g., context) that could provide additional insights into whether the content is hateful. Although both HateBERT and HateXplain have some overlap, there is very little overlap across some platform combinations. For instance, both these models have little overlap in terms of representations for GAB and YouTube, whereas both Reddit and Twitter have high overlap.

4.5 Contribution of the Components (RQ.3)

The **CATCH** architecture is a complex construct composed of various integrated modules and losses. To assess the influence of these individual elements on cumulative performance, we have conducted an experiment featuring four distinct variations of the **CATCH**. These variations include: (i) *CATCH w/o Hate & Target Loss*, which excludes hate and target losses, (ii) *CATCH w/o finetuning*, which omits the fine-tuning for the LM encoder and decoder, (iii) *CATCH w/o Hate Loss*, which forgoes the Hate Loss, and (iv) *CATCH w/o Target Loss*, which disregards the Target Loss.

We trained these variations using the Reddit and Twitter datasets in cross-platform experiments. The outcomes of these experiments are illustrated in Figure 5(a) for the Reddit dataset and Figure 5(b) for the Twitter dataset. The results demonstrate that the **CATCH**



(a) Reddit

(b) Twitter

Figure 5: Macro-F1 score reflecting the importance of each cue compared with the final model for Reddit and Twitter.

delivers optimal performance when both the fine-tuning process and the various loss components are incorporated. Neglecting these components results in a 10-18% performance drop.

Among the variations, the most significant performance benefit for the **CATCH** comes from including target and hate losses. These losses act as conduits to ensure that the disentangled components encapsulate information exclusive to their counterparts. By doing so, the disentanglement is effectively guided, resulting in generalizable representations. The performance significantly deteriorates when the hate loss or target loss is omitted. In addition, as the LM encoder and decoder are not trained on hateful content, the absence of any fine-tuning also leads to a significant performance decline.

Models	Target Platforms			
	GAB	Reddit	Twitter	YouTube
GPT4	**0.64**	0.66	**0.67**	0.63
Falcon	0.42	0.58	0.54	0.55
CATCH (Avg.)	0.61	**0.71**	0.64	**0.70**

Table 3: Performance comparison of LLMs, GPT4 and Falcon, with CATCH for generalizable hate speech detection.

5 ARE LLMS GENERALIZABLE WHEN IT COMES TO HATE SPEECH? – A CASE STUDY

Using large language models (LLMs), the NLP community has significantly improved language comprehension in recent years. Due in large part to their ability to process enormous volumes of data and their vast model capacity, these models have shown excellent performance across a variety of natural language processing applications [22–24]. Yet, whether these LLMs generalize effectively for complex tasks such as hate speech remains unexplored.

Despite their outstanding performance across a wide range of domains, it's critical to assess how well these LLMs can be applied to difficult real-world issues, particularly those that rely on comprehension of complicated social settings, such as hate speech. Given that these models have significant modeling capabilities and have been trained on large datasets [25, 34], it is logical to assume that they can generalize effectively. However, to test this hypothesis, we conducted a performance comparison of **CATCH** with the leading-edge LLMs, GPT4 [33] and Falcon [34], as shown in Table 3. We give the following prompt to GPT4 and Falcon, and note the label predictions generated by these models. We provide a few instances of hateful and non-hateful content from each platform to leverage the model's capabilities to easily do the prediction. The outcomes show how well the models generalizes across the platforms.

Prompt to Detect Hate

Here are some examples of hateful content on `Platform`: `samples`. Based on the given examples, decide whether the following text is hateful or not? Just answer in Yes or No.

According to the findings, the GPT4 model performs well on two platforms. This can be attributed to its intensive training on a wide range of data and enormous model capacity. However, its performance deteriorates with both Reddit and YouTube. On the other hand, Falcon performs a little worse, with scores ranging from 0.42 on GAB to 0.58 on Reddit. The average cross-platform performance of our suggested **CATCH** significantly improves, with the highest ratings on Reddit (0.71) and YouTube (0.7).

In conclusion, although LLMs like GPT4 and Falcon have potential, they may fall short in nuanced tasks like hate speech detection. **CATCH**, however, uses causality-aware disentanglement, making it more effective in dealing with platform-specific variations in hate.

6 CONCLUSION AND FUTURE WORK

Targeted hate speech, a growing issue on social media platforms, necessitates the development of effective automated detection techniques. Addressing the challenges of hate speech proliferation and scarce labeled data, our study presents the **CATCH**, a novel framework that disentangles causal (platform invariant) and target (platform-dependent) components from input texts for enhanced hate speech detection. By separating platform-dependent traits, the invariant component aids successful generalization across diverse platforms. Experiments confirmed **Model's** superiority over existing baselines and verified the invariant nature of its learned causal representations. Additional case studies revealed insights into the performance of large-scale language models like GPT4 and Falcon in generalizable hate speech detection.

However, **CATCH**'s reliance on target labels for representation segregation poses a challenge, given that such labels may not always be available. Future work could focus on enhancing the **CATCH** framework to operate effectively without target labels. The insights and results from this study lay a promising groundwork for creating a safer and more respectful digital social environment.

7 ETHICAL STATEMENT

7.1 Freedom of Speech and Censorship

Our research aims to develop algorithms that can effectively identify and mitigate harmful language across multiple platforms. We recognize the importance of protecting individuals from the adverse effects of hate speech and the need to balance this with upholding free speech. Content moderation is one application where our method could help censor hate speech on social media platforms such as Twitter, Facebook, Reddit, etc. However, one ethical concern is our system's false positives, i.e., if the system incorrectly flags a user's text as hate speech, it may censor legitimate free speech. Therefore, we discourage incorporating our methodology in a purely automated manner for any real-world content moderation system until and unless a human annotator works alongside the system to determine the final decision.

7.2 Use of Hate Speech Datasets

In our work, we incorporated publicly available well-established datasets. We have correctly cited the corresponding dataset papers and followed the necessary steps in utilizing those datasets in our work. We understand that the hate speech examples used in the paper are potentially harmful content that could be used for malicious activities. However, our work aims to help better investigate and help mitigate the harms of online hate. Therefore, we have assessed that the benefits of using these real-world examples to explain our work better outweigh the potential risks.

7.3 Fairness and Bias in Detection

Our work values the principles of fairness and impartiality. To reduce biases and ethical problems, we openly disclose our methodology, results, and limitations and will continue to assess and improve our system in the future.

REFERENCES

[1] Davut Akca, Fatih Karakus, Mehmet F Bastug, and Barbara Perry. 2020. Planting hate speech to harvest hatred: How does political hate speech fuel hate crimes in Turkey? *International Journal for Crime, Justice and Social Democracy* 9, 4 (2020), 195–211.

[2] Dimosthenis Antypas and Jose Camacho-Collados. 2023. Robust Hate Speech Detection in Social Media: A Cross-Dataset Empirical Evaluation. *arXiv preprint arXiv:2307.01680* (2023).

[3] Tom Bourgeade, Patricia Chiril, Farah Benamara, and Véronique Moriceau. 2023. What Did You Learn To Hate? A Topic-Oriented Analysis of Generalization in Hate Speech Detection. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational LinguisticsProceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. 3477–3490.

[4] Peter Bühlmann. 2020. Invariance, causality and robustness. (2020).

[5] Paula Carvalho, Danielle Caled, Cláudia Silva, Fernando Batista, and Ricardo Ribeiro. 2023. The expression of hate speech against Afro-descendant, Roma, and LGBTQ+ communities in YouTube comments. *Journal of Language Aggression and Conflict* (2023).

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

[7] Ke-Li Chiu, Annie Collins, and Rohan Alexander. 2021. Detecting hate speech with gpt-3. *arXiv preprint arXiv:2103.12407* (2021).

[8] 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 (2023), 119446.

[9] Nicola Döring and M Rohangis Mohseni. 2019. Male dominance and sexism on YouTube: results of three content analyses. *Feminist Media Studies* 19, 4 (2019), 512–524.

[10] 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*, Vol. 12.

[11] Tracie Farrell, Miriam Fernandez, Jakub Novotny, and Harith Alani. 2019. Exploring misogyny across the manosphere in reddit. In *Proceedings of the 10th ACM conference on web science*. 87–96.

[12] Hao-Zhe Feng, Kezhi Kong, Minghao Chen, Tianye Zhang, Minfeng Zhu, and Wei Chen. 2021. Shot-vae: semi-supervised deep generative models with label-aware elbo approximations. In *Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence Conference on Artificial Intelligence*, Vol. 35. 7413–7421.

[13] Paula Fortuna, Juan Soler, and Leo Wanner. 2020. Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In *LRECProceedings of the 12th language resources and evaluation conference*. 6786–6794.

[14] Paula Fortuna, Juan Soler-Company, and Leo Wanner. 2021. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management* 58, 3 (2021), 102524.

[15] Tanmay Garg, Sarah Masud, Tharun Suresh, and Tanmoy Chakraborty. 2022. Handling bias in toxic speech detection: A survey. *Comput. Surveys* (2022).

[16] Md Saroar Jahan and Mourad Oussalah. 2023. A systematic review of Hate Speech automatic detection using Natural Language Processing. *Neurocomputing* (2023), 126232.

[17] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).

[18] Ujun Jeong, Paras Sheth, Anique Tahir, Faisal Alatawi, H Russell Bernard, and Huan Liu. 2023. Exploring Platform Migration Patterns between Twitter and Mastodon: A User Behavior Study. *arXiv preprint arXiv:2305.09196* (2023).

[19] 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* (2020).

[20] 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*. 6667–6679.

[21] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. *Advances in neural information processing systems* 27 (2014).

[22] 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* (2023), 101861.

[23] Anis Koubaa. 2023. GPT-4 vs. GPT-3.5: A concise showdown. (2023).

[24] Xianzhi Li, Xiaodan Zhu, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2023. Are ChatGPT and GPT-4 General-Purpose Solvers for Financial Text Analytics? An Examination on Several Typical Tasks. *arXiv preprint arXiv:2305.05862* (2023).

[25] Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian, Hao He, Antong Li, Mengshen He, Zhengliang Liu, et al. 2023. Summary of chatgpt/gpt-4 research and perspective towards the future of large language models. *arXiv preprint arXiv:2304.01852* (2023).

[26] Divyat Mahajan, Shruti Tople, and Amit Sharma. 2021. Domain generalization using causal matching. In *International Conference on Machine Learningnternational Conference on Machine Learning*. PMLR, 7313–7324.

[27] Jitendra Singh Malik, Guansong Pang, and Anton van den Hengel. 2022. Deep learning for hate speech detection: a comparative study. *arXiv preprint arXiv:2202.09517* (2022).

[28] Zainab Mansur, Nazlia Omar, and Sabrina Tiun. 2023. Twitter Hate Speech Detection: A Systematic Review of Methods, Taxonomy Analysis, Challenges, and Opportunities. *IEEE Access* (2023).

[29] 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*. 149–159.

[30] Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*. 173–182.

[31] 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 intelligenceProceedings of the AAAI conference on artificial intelligence*, Vol. 35. 14867–14875.

[32] Changrong Min, Hongfei Lin, Ximing Li, He Zhao, Junyu Lu, Liang Yang, and Bo Xu. 2023. Finding hate speech with auxiliary emotion detection from self-training multi-label learning perspective. *Information Fusion* 96 (2023), 214–223.

[33] OpenAI. 2023. GPT-4 Technical Report. *ArXiv* abs/2303.08774 (2023).

[34] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei,

and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116* (2023). arXiv:2306.01116 https://arxiv.org/abs/2306.01116

[35] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277* (2023).

[36] Raúl Pérez. 2017. Racism without hatred? Racist humor and the myth of "color-blindness". *Sociological Perspectives* 60, 5 (2017), 956–974.

[37] Alan Ramponi and Sara Tonelli. 2022. Features or Spurious Artifacts? Data-centric Baselines for Fair and Robust Hate Speech Detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language TechnologiesProceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 3027–3040. https://doi.org/10.18653/v1/2022.naacl-main.221

[38] Sumegh Roychowdhury and Vikram Gupta. 2023. Data-Efficient Methods For Improving Hate Speech Detection. In *Findings of the Association for Computational Linguistics: EACL 2023*. 125–132.

[39] 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 *ICWSM'18Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 12.

[40] 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*. 1–10.

[41] Paras Sheth, Tharindu Kumarage, Raha Moraffah, Aman Chadha, and Huan Liu. 2023. PEACE: Cross-Platform Hate Speech Detection-A Causality-guided Framework. *arXiv preprint arXiv:2306.08804* (2023).

[42] Paras Sheth, Raha Moraffah, K Selçuk Candan, Adrienne Raglin, and Huan Liu. 2022. Domain Generalization–A Causal Perspective. *arXiv preprint arXiv:2209.15177* (2022).

[43] Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*. 138–142.

[44] Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*. 88–93.

[45] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771* (2019).

[46] 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* (2022).

[47] Xuhui Zhou. 2021. *Challenges in automated debiasing for toxic language detection*. University of Washington.