Intent-Aware and Hate-Mitigating Counterspeech Generation via Dual-Discriminator Guided LLMs

Haiyang Wang[1], **Zhiliang Tian**[1]*, **Xin Song**[1] **Yue Zhang**[1]
Yuchen Pan[2], **Hongkui Tu**[1], **Minlie Huang**[3], **Bin Zhou**[1]
[1]National University of Defense Technology
[2]Academy of Military Sciences
[3]Tsinghua University
{wanghaiyang19, tianzhiliang, xinsong, zhangyue, yuchenpan, tuhongkui, binzhou}@nudt.edu.cn
aihuang@tsinghua.edu.cn

Abstract

Counterspeech is an effective way to combat online hate speech. Considering the multifaceted nature of online hate speech, counterspeech with varying intents (e.g., denouncing or empathy) has significant potential to mitigate hate speech effectively. Recently, controlled approaches based on large language models (LLMs) have been explored to generate intent-specific counterspeech. Due to the lack of attention to intent-specific information by LLMs during the decoding process, those methods cater more to the semantic information rather than matching with the desired intents. Further, there are still limitations in quantitatively evaluating the effectiveness of counterspeech with different intents in mitigating hate speech. In this paper, to address the above issues, we propose **DART**, an LLMs-based **D**u**A**l-disc**R**imina**T**or guided framework for counterspeech generation. We employ an intent-aware discriminator and hate-mitigating discriminator to jointly guide the decoding preferences of LLMs, which facilitates the model towards generating counterspeech catering to specific intent and hate mitigation. We apply a maximum-margin relative objective for training discriminators. This objective leverages the distance between counterspeech aligned with the desired target (such as specific intent or effectiveness in hate mitigation) and undesired as an effective learning signal. Extensive experiments show that DART achieves excellent performances in matching the desired intent and mitigating hate.

Keywords: Large Language Models, Counterspeech Generation, Controllable Text Generation

1. Introduction

Hate speech (**HS**) is an aggressive expression that incites hatred towards specific groups based on their group identity (religion, ethnicity, nationality or race etc.) (Nockleby, 2000). The widespread spread of HS on social media has made communication more aggressive. One effective way to combat online hate is through counterspeech (**CS**), which involves directly responding to HS to reduce its negative impact and promote a more friendly and harmonious dialogue. Recently, many non-governmental organizations (NGOs)[1][2] have recruited volunteers to manually write CS to combat HS. However, owing to the vast amount of HS generated on the web daily, automatically generating CS may be a better approach to minimize human intervention.

Recently, much research has focused on CS generation. These research can be divided into two categories: non-controllable and controllable approaches. Non-controllable methods (Qian et al., 2019; Zhu and Bhat, 2021; Saha et al., 2022) usually generate one CS that is most similar to the HS in semantic. They also tend to generate CS that is similar to the gold CS in the training dataset, resulting in a lack of diversity in style. A single-style CS may not have a good effect on mitigating hatred (Gupta et al., 2023). Controllable approaches (Chung et al., 2021; Krause et al., 2021) can generate informative or positive CS by utilizing knowledge-control or emotional-control techniques. However, these methods often prioritize a specific intent, overlooking the potential for CS with diverse intents such as humor, denunciation, and empathy (Gupta et al., 2023; Mathew et al., 2019). This limitation stems from the specialized nature of their model architecture, which hinders the generation of CS with varying intents.

Numerous studies (Mathew et al., 2019; Benesch et al., 2016; Hangartner et al., 2021) have shown that using **diverse CS with various intents**, designed for different scenarios, holds great potential in effectively combating HS. Gupta et al. (2023) proposed the intent-specific counterspeech generation task, which aims to generate a CS for a given HS and a desired CS intent. They propose a two-phased counterspeech generation framework, namely QUARC. It learns vector-quantized representations for each intent in the first stage and utilizes these learned representations to generate intent-specific CS in the second stage. QUARC achieves superior performance in intent-

* Corresponding authors
[1]https://www.wecounterhate.com/
[2]https://getthetrollsout.org/

LREC-COLING 2024, pages 9131–9142
20-25 May, 2024.

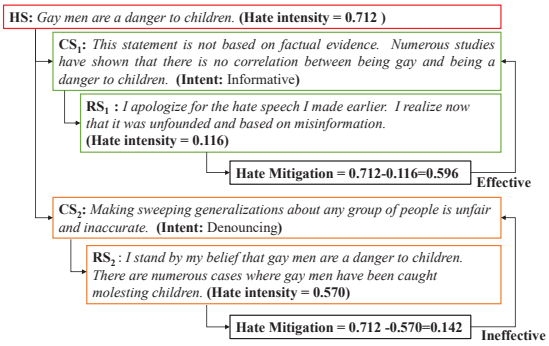

Figure 1: The effectiveness of CS against HS can be judged based on the hate mitigation value.

specific counterspeech generation tasks. However, QUARC is still based on BART which may cause the detected intent and the desired intent to be inconsistent. Recently, large language models (LLMs) such as GPT4 (OpenAI, 2023; Zhong et al., 2023) and LLaMa (Touvron et al., 2023), have demonstrated strong instruction understanding and text generation abilities. Therefore, utilizing instruction prompts to constrain the probability distribution of LLMs and provide supervised signals about the desired intent during the decoding process could be a more effective approach.

The primary objective of CS is to **mitigate hate and decrease conflict** in conversations (Benesch et al., 2016; ling Chung et al., 2023). CS with various intents may have varying effectiveness in mitigating hate. The aforementioned methods primarily focus on enhancing the generation quality of CS in qualitative aspects (e.g., fluency or diversity) to mitigate hate. However, there has been limited quantitative evaluation of the effectiveness of generated CS in mitigating hate during conversations. Therefore, when generating CS, it is also necessary to guide towards effective hate mitigation. The effectiveness of CS can be quantitatively evaluated by the difference in hate intensity (referred to as the hate mitigation value) between HS and the responses (RS) to CS (Garland et al., 2022; Chung et al., 2023). The greater the difference in hate intensity, the more effective the CS is considered to be. As the example in Figure 1, the hate mitigation value of CS_1 is greater than that of CS_2, so CS_1 is considered a more effective counterspeech against hate speech.

In this paper, we propose **DART**, an LLM-based **D**u**A**l-disc**R**imina**T**or guided framework for counterspeech generation, which learns to mitigate the hate of online speech catering to the corresponding intents. Firstly, to obtain intent-aware CS via LLMs, we construct intent-aware prompts for LLMs considering the target hate speech and the given intent. Secondly, we propose two sub-sequence level discriminators to dynamically provide step-level feedback for LLMs while the LLMs' generation: (1) an intent-aware discriminator to check whether the generated CS aligns with the desired intent. (2) a hate mitigation discriminator to estimate the ability of the generated CS relieving the hate intensity of the user's next response. To train the discriminator, we design a maximum-margin relative objective by maximizing the gap between CS with desired attributes (*specific intent* or *effective in hate mitigation*) and undesired attributes. Finally, the framework employs a dual-discriminator guided decoding module to jointly and dynamically adjust the decoding preferences of LLMs and nudge the model towards generating CS with desired attributes. Our contributions are threefold:

- We propose an LLM-based dual-discriminator guided framework for counterspeech generation. It learns to mitigate the hate of online speech catering to the desired intent.
- We design a maximum-margin relative objective to train the intent-aware and hate-mitigating discriminator by leveraging the distance between counterspeech aligned with the desired attributes and undesired as an efficient learning signal.
- Extensive experiments show that DART achieves excellent performances in matching the desired intent and mitigating hate.

2. Related Work

2.1. Counterspeech Generation

Counterspeech (**CS**) can be defined as a direct response to hate or dangerous speech to mitigate hate. CS can fight hate speech (**HS**) and reduce its negative impact on social media while still allowing free speech (ling Chung et al., 2023). Many researchers are working to provide high-quality training data for CS generation models. Qian et al. (2019) conducted an initial attempt to construct the HS-CS dataset by employing crowdsourced workers. Subsequently, the CONAN dataset series was introduced which contains CONAN (Chung et al., 2019), Multi-target CONAN (Fanton et al., 2021), KCONAN (Chung et al., 2021), DIALOCONAN (Bonaldi et al., 2022). Recently, (Gupta et al., 2023) proposed the Intent-CONAN dataset, which provides an intent label for each CS.

Based on these datasets, many automatic CS generation systems are proposed. Qian et al. (2019) use Seq2Seq and VAE as baselines for generative CS. Zhu and Bhat (2021) propose Generate-Prune-Select which is a three-stage pipeline to obtain the most relevant CS for an HS instance. Chung et al. (2021) proposed a knowledge-grounded generation approach by incorporating an

intermediate step in which keyphrases are generated to retrieve the necessary knowledge. Saha et al. (2022) proposed CounterGEDI, an ensemble of GEDI to guide the generation of a DialoGPT model toward more polite, detoxified, and emotional CS. Then, Gupta et al. (2023) propose QUARC, which leverages vector-quantized representations learned for each intent category along with a fusion module to incorporate them into the model. These models primarily guide the generation of specific CS from a qualitative perspective (e.g. Empathetic CS usually is effective (Hangartner et al., 2021).), and have limitations in terms of quantitatively reducing hate intensity.

2.2. Controllable Text Generation

The controllable text generation task aims to generate text with specified attributes, such as sentiment and topic, while allowing researchers to control the generated output(Yang et al., 2023; Zhang et al., 2022). Existing works include (1) fine-tuning pre-trained language models (PLMs), (2) using additional attribute discriminators, and (3) prompt-based approaches. For the first type of approach, a variety of methods have been proposed. For instance, StylePTB (Lyu et al., 2021) propose a fine-grained controllable approach for text style transfer. GSum(Dou et al., 2021) introduces four types of guidance signals to enhance the controllability of PLMs and generate more faithful summaries. These researches have propelled the advancement of controllable text generation. However, the cost of retraining PLMs is enormous. Therefore, some researchers have redirected their attention towards developing flexibility and plug-and-play methods.

The second type of approach utilizes extra attribute discriminators to guide PLMs during token generation. PPLM(Dathathri et al., 2020) progressively modifies the latent representations of a GPT-2 by referencing the gradient of attribute classifiers. Fudge(Yang and Klein, 2021) employs an attribute predictor to fine-tune the output probabilities of a PLM. GeDi(Krause et al., 2021) and DExperts(Liu et al., 2021a) utilize class-conditioned language models for positive and negative classes.

Lately, with the continued advancement of LLMs, prompt-based approaches(Liu et al., 2023; Gao et al., 2024) have garnered the interest of scholars. These approaches utilize discrete (Brown et al., 2020; Zou et al., 2021; Zhong et al., 2022) or continuous prompts (Liu et al., 2021b; Yang et al., 2023) to control language models towards generating specific content. The aforementioned models have exhibited encouraging outcomes for controlled generation and possess the potential to aid in generating intent-specific CS against HS. Nevertheless, the generated CS still faces restrictions in effectively diminishing hate and precisely aligning with intents.

3. Method

3.1. Task Description

Formally, there is a set of hate speech(**HS**) instances with counterspeech(**CS**) towards $\mathcal{S} = \{(x_1, c_1, y_1), \cdots, (x_N, c_N, y_N)\}$, where x_i is the ith HS instances, y_i is the CS corresponding to x_i, and c_i is the desired intent. The goal of the intent-specific CS generation task is to construct a stochastic text generation function χ. It can take HS x_i and desired intent c_i as the input and output the generated CS $\hat{y}_i$. The semantic of it is related to the x_i and the form in line with the c_i, such that $\hat{y}_i \sim \chi(\cdot|x_i, c_i)$.

3.2. Model Architecture

Our framework DART, is based on an LLM and two discriminators. As shown in Figure 2, it consists of four modules:

- **Intent-aware Prompt Constructor** constructs an instruction prompt for LLMs considering the target hate speech and the given intents.
- **Sub-sequence Level Hate Mitigation Discriminator** judges whether a CS (or a sub-sequence of CS) can relieve the hate intensity of the user's next response, where achieved by using the maximum-margin relative objective to maximize the gap between effective sub-sequence and ineffective. This module step-by-step guides the LLM to generate a CS to mitigate the hate intensity.
- **Sub-sequence Level Intent-aware Discriminator** calculate the probability that a given CS (or a sub-sequence of CS) matches the desired intent. This module guides the LLM to generate intent-specific CS.
- **Dual-discriminator Guided LLM Decoding Module** takes instruction prompts from the prompt generator as inputs to the LLM and then uses the hate mitigation and intent-aware discriminators to jointly guide the decoding process for generating CS. The two discriminators provide feedback to the LLM at each generation step.

3.3. Intent-aware Prompt Constructor

This module mainly focuses on generating an instruction prompt that integrates the HS that needs to be countered and the desired intent. Then, it can prompt LLM to generate an intent-specific CS to respond to the HS. Specifically, we design a prompt template $ppt(\cdot, \cdot)$ which takes the HS x and the desired intent c as input. The template is "*Given the hate speech: {HS}, please generate a counter-speech utilizing the {INTENT} approach.*"

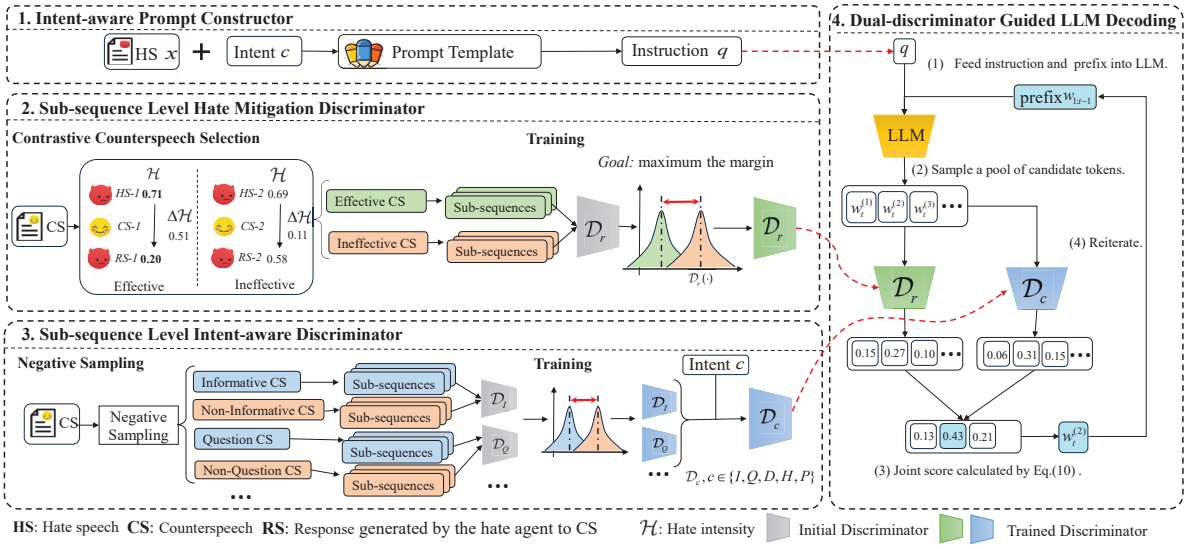

Figure 2: The architecture of DART. 1. Intent-aware prompt constructor constructs an instruction prompt for LLM. 2. Sub-sequence level hate mitigation discriminator judges whether a subsequence can effectively mitigate hate. 2. Sub-sequence level intent-aware discriminator calculates the probability that a given subsequence matches the desired intent. 4. Dual-discriminator guided LLM decoding module uses the trained two discriminators to jointly guide the decoding process of LLM.

Then, we can obtain an instruction prompt $q = ppt(x, c)$ to prompt LLM to generate a CS $\hat{y} = \{w_1, \cdots, w_n\}$ token-by-token, where n is the total number of tokens and w_t is the t-th token of $\hat{y}$.

3.4. Sub-sequence Level Hate Mitigation Discriminator

This module aims to train the hate mitigation discriminator $\mathcal{D}_r$ to improve its ability to judge whether it can alleviate hate intensity. The module consists of two sub-modules: (1) **Contrastive counterspeech selection** sub-module is to select CS instances from the training dataset that can effectively mitigate hate and those that cannot; (2) **Maximum-Margin Relative Learning** sub-module aims to train the hate mitigation discriminator based on a *maximum-margin relative objective*, to identify subsequences that can effectively mitigate hate.

3.4.1. Contrastive Counterspeech Selection

This sub-module selects a set with effective CS (i.e. instances that effectively mitigate hate) and a set with ineffective CS that hardly mitigates hate, where we introduce a hate agent $M_{\mathcal{H}}$ to separate the training set through three phases.

Response generation phase. We apply an uncensored LLM as hate agent $M_{\mathcal{H}}$ and it can take HS x and the CS y as input. We use a prompt to make $M_{\mathcal{H}}$ mimic the speaker of the HS x, and then the model generates N_z possible responses z to the CS y. This process can be formalized as $\{z_j\}_{j=1}^{N_z} = M_{\mathcal{H}}(x, y)$

Hate mitigation calculation phase. We define a hate intensity function $\mathcal{H}(\cdot)$ following (Sahnan et al., 2021; Dahiya et al., 2021). It is based on a model-dependent score $\mathcal{C}(\cdot)$ and a lexicon-based score $Le(\cdot)$. First, we employ a state-of-the-art HS classifier $\mathcal{C}(\cdot)$ which classifies a text into HS with a score, indicating the probability of the text being an HS. Then, we can obtain a model-independent lexicon-based hate score $Le(\cdot)$ based on a domain-independent hate lexicon (Wiegand et al., 2018) with 2,895 hate words. Wiegand et al. (2018) assigned hate score to each word in the lexicon. We examine the presence of hate words in a text and sum their hate score. The hate intensity function $\mathcal{H}(\cdot)$ can be defined as

$$\mathcal{H}(\cdot) = \gamma \mathcal{C}(\cdot) + (1 - \gamma) Le(\cdot) \quad (1)$$

where γ adjusts the weights of two components. After that, we quantify the hate mitigation effect of the CS y by measuring the difference $\Delta\mathcal{H}$ in hate intensity between HS x and responses z generated by the hate agent $M_{\mathcal{H}}$ as follows:

$$\Delta\mathcal{H} = \mathcal{H}(x) - \frac{1}{N_z} \sum_{j=1}^{N_z} \mathcal{H}(z_j) \quad (2)$$

we first calculate the average hate intensity of all responses. Then the hate intensity of the HS subtracts the average.

Contrastive sample selection phase. We define two thresholds θ^+ and θ^- for contrastive counterspeech selection. As for a CS y_i, if $\Delta\mathcal{H}_i > \theta^+$,

then we annotate it as an effective instance. If $\Delta\mathcal{H}_i < \theta^-$, we regard it is ineffective.

After completing the above three phases, we obtain two sets of CS instances, one that is effective and the other is ineffective in mitigating hate. To assist the training of the hate mitigation discriminator, we form a tuple of CS y_i and its corresponding instruction q_i. We can then obtain the effective tuple set $\mathcal{S}_e = \{(q_i, y_i)\}_{i=1}^{N_e}$ and ineffective tuple set $\mathcal{S}_{e^-} = \{(q_j, y_j)\}_{j=1}^{N_{e^-}}$.

3.4.2. Maximum-Margin Relative Learning (MMRL)

To help the discriminator perform on the sub-sequence level, we construct sub-sequence level effective and ineffective CS sets and propose a maximum-margin relative objective to train the hate mitigation discriminator $\mathcal{D}_r$. Specifically, the discriminator model $\mathcal{D}_r$ takes in the instruction q, the prefix $\{w_1, w_2, \cdots, w_{t-1}\}$ and a candidate next token w_t. And it outputs a real-valued score $\mathcal{D}_r(q, w_{1:t-1}, w_t)$ indicates whether w_t is an effective token at time-step t under the condition the prefix $w_{1:t-1}$ and instruction q. To cater to the maximum-margin relative objective, we use two following phases to learn the discriminator function $\mathcal{D}_r(q, w_{1:t-1}, w_t)$.

Sample construction. We construct sub-sequence level effective CS and ineffective CS set by cutting out prefix text spans from the sentence-level CS sets $\mathcal{S}_e = \{(q_i, y_i)\}_{i=1}^{N_e}$ and $\mathcal{S}_{e^-} = \{(q_j, y_j)\}_{j=1}^{N_{e^-}}$ (obtained in the section 3.4.1). We use the effective tuple sets $\mathcal{S}_e = \{(q_i, y_i)\}_{i=1}^{N_e}$ as an example. As previously mentioned, y_i can be represented as $\{w_1^i, \cdots, w_{n_i}^i\}$, then we can construct a separate example (Yang and Klein, 2021) for each prefix $w_{1:t-1}^i$ of $w_{1:n_i}^i$. It results a set of triplets $\mathbf{T}_e^i = \{(q_i, w_{1:t-1}^i, w_t^i)\}_{t=2}^{n_i}$. Therefore, we can get $\mathcal{S}_e = \{\mathbf{T}_e^i\}_{i=1}^{N_e}$ and $\mathcal{S}_{e^-} = \{\mathbf{T}_{e^-}^j\}_{j=1}^{N_{e^-}}$.

Training with MMRL. To enlarging the margin of the real-value score between effective and ineffective instances, we design a novel maximum-margin relative objective $\mathcal{L}_{MMR}$ based on previous research (Liu et al., 2020; Li et al., 2020; Khalifa et al., 2023) to train the discriminator $\mathcal{D}_r$. It can be defined as:

$$\mathcal{L}_{MMR} = -(\delta_e - \delta_{e^-}) + \lambda \tag{3}$$

$$\delta_e = \frac{1}{N_e \cdot n_i} \sum_{i=1}^{N_e} \sum_{t=2}^{n_i} \left(\mathcal{D}_r(q_i, w_{1:t-1}^i, w_t^i)\right) \tag{4}$$

$$\delta_{e^-} = \frac{1}{N_{e^-} \cdot n_j} \sum_{j=1}^{N_{e^-}} \sum_{t=2}^{n_j} \left(\mathcal{D}_r(q_j, w_{1:t-1}^j, w_t^j)\right) \tag{5}$$

where $\lambda > 0$ is the margin hyperparameter. Intuitively, the relativistic relation shows the gap between the real-value of the effective samples and ineffective samples. We aim to train the discriminator to widen the gap, thereby enhancing its discriminative ability.

3.5. Sub-sequence Level Intent-aware Discriminator

We train a discriminator to model the probability that a sub-sequence matches the desired intent. It consists of two sub-modules: (1) **Negative sampling** sub-module reorganizes the training dataset into five sets based on intent. Each group contains CS instances with desired intent and CS with undesired intent; (2) **Intent-aware Discriminator training** sub-module trains five different discriminators using the maximum-margin relative learning, to distinguish whether a sub-sequence matches the desired intent.

3.5.1. Negative sampling

The purpose of this sub-module is to collect negative samples for CS instances with each intent. Firstly, we divide the training dataset into five sets $\{\mathcal{S}_I, \mathcal{S}_D, \mathcal{S}_P, \mathcal{S}_Q, \mathcal{S}_H\}$, according to intents {*Informative, Denonncing, Positive, Question, Humor*}. Then, we collect a set of negative samples for each intent set from the other four different sets, which can be represented as $\{\mathcal{S}_{I^-}, \mathcal{S}_{D^-}, \mathcal{S}_{P^-}, \mathcal{S}_{Q^-}, \mathcal{S}_{H^-}\}$. It is worth noting that due to the uneven number of instances for each intent, there is an issue of imbalanced data distribution. Therefore, to ensure (1) balance between desired intent and undesired intent and(2) balance various undesired intents in the negative set, we performed random under-sampling for intents with many instances and over-sampling for intents with too few instances.

3.5.2. Intent-aware Discriminator Training

In this sub-module, our goal is to train a discriminator for each intent using the maximum-margin relative objective. We also construct each sample following with sample construction phase in section 3.4.2. Then, based on the pairwise sets created earlier, we can obtain a set of intent-specific discriminators, denoted as $\mathcal{D}_c, c \in \{I, D, P, Q, H\}$. We can then select the appropriate discriminator based on the desired intent c.

3.6. Dual-discriminator Guided LLM Decoding Module

We use two discriminators to jointly guide the decoding of LLM to make it consider both hate mitigation and intent-aware. Specifically, an LLM distribution with the instruction q can be defined as $p_{LLM}(\cdot \mid q)$. It can generate w_t followed by $p_{LLM}(w_t \mid q, w_{1:t-1})$.

And the complete CS $\hat{y}$ can be generated by following the factoring:

$$p_{LLM}(\hat{y} \mid q) = \prod_{t=1}^{n} p_{LLM}(w_t \mid q, w_{1:t-1}). \quad (6)$$

Once $\mathcal{D}_r$ and $\mathcal{D}_c$ have been trained, they can be utilized to guide the CS generation. For each time t, we can sample a token pool $\mathcal{S}_t = \{w_t^{(1)}, w_t^{(2)}, \cdots, w_t^{(N_t)}\}$ from the LLM distribution $p_{LLM}(w_i \mid q, w_{1:t-1})$. The token pool consists of N_t candidates. We choose the most appropriate candidate based on the comprehensive score $s_t^{(i)}$, which can be calculated using the following formula:

$$\begin{aligned} s_t^{(i)} =& \gamma_{LM} log[p_{LM}(w_t^{(i)} \mid q, w_{1:t-1})] \\ &+ \gamma_r \hat{\mathcal{D}}_r(q, w_{(1:t-1)}) + \gamma_c \hat{\mathcal{D}}_c(q, w_{(1:t-1)}) \end{aligned} \quad (7)$$

where $\gamma_{LM}, \gamma_r, \gamma_c$ are controllable hyperparameter bias the generation toward the desired intent c and effectively reduce the hate intensity.

We normalize the initial discriminator score $\mathcal{D}(q, w_{1:t-1})$ to obtain the normalized score $\hat{\mathcal{D}}(q, w_{1:t-1})$. After selecting the most suitable token at time step t, we add it to the prefix and continue the iterative process until the final CS is generated. During this process, both discriminators participate jointly in each iteration to ensure the controllability of the decoding process.

In summary, we can sample the token w_t that satisfies the following criteria: **(i)** it has a high likelihood $p_{LLM}(w_t \mid q, w_{1:t-1})$ according to the LLM. **(ii)** it has a large potential to effectively reduce the hate intensity. **(iii)** it aligns with the desired intent c with a high likelihood. This ensures that the generated token is not only grammatically and semantically correct but also appropriate in terms of reducing hate intensity and fulfilling the desired CS intent c.

4. EXPERIMENTS

4.1. Datasets

We created the ICONAN dataset based on the construction process of IntentCONAN(Gupta et al., 2023). The ICONAN dataset is created based on CONAN (Chung et al., 2019) and Multi-Target CONAN (Fanton et al., 2021) dataset. As for CONAN, we club some semantically similar intents together of CONAN. We consider five intent categories, i.e., *informative*, *question*, *denouncing*, *humor*, and *positive* in ICONAN. Then, we annotate the intent label for CS of Multi-Target CONAN with the help of ChatGPT by constructing an annotation prompt based on the definition of intent. We combine the adjusted CONAN and Multi-Target CONAN datasets to create ICONAN. The detailed statistical information is shown in Table 1. We split the dataset into 70%/15%/15% for training, validation, and testing.

Hate Speech		Counterpeech					
Targets	**Counts**	**INF**	**DEN**	**POS**	**QUE**	**HUM**	**Total**
Musilim	1316	2655	1837	747	659	389	6287
Migrants	634	621	22	254	48	2	947
Women	560	346	78	202	36	0	662
LGBT+	465	277	66	244	29	1	617
Jews	418	376	67	97	45	9	594
POC	301	183	61	81	27	0	352
Disabled	175	100	10	104	6	0	220
Other	181	159	24	71	24	0	268
Total	**4050**	**4717**	**2165**	**1800**	**864**	**401**	**9947**
Train	3071	3301	1515	1260	605	281	6962
Dev	948	708	325	270	129	60	1492
Test	982	708	325	270	130	60	1493

Table 1: Dataset Statistics of ICONAN.

4.2. Evaluation Metrics

We evaluated the generated CS from multiple perspectives. Specifically, **Semantic Similarity** aims to evaluate the semantic consistency. **Novelty** aims to examine the difference between the generated CS and the training corpus. **Diversity** aims to determine if the generator can produce diverse sentences. **Toxicity** indicates whether the generated CS can be considered toxic. **Politeness** indicates whether the generated CS is polite. **Intent Accuracy** determines if the CS adheres to the desired intent. **Hate Mitigation** aims to evaluate the effectiveness of the CS in mitigating hate in a conversation. Details are as follows:

- **Semantic similarity (SS)**: Following (Gupta et al., 2023), we also report the semantic similarity obtained from a sentence-transformers model which is *all-miniLM-v2.*
- **Novelty (N)**: We calculate the novelty(Wang and Wan, 2018) of each generated CS $\hat{y}_i$ using the following formula:

$$\text{Novelty}\,(\hat{y}_i) = 1 - \max\{\varphi\,(\hat{y}_i, y_j)\}_{j=1}^{j=|Y|} \quad (8)$$

  $\varphi(\cdot,\cdot)$ is Jaccard similarity function, Y is the CS set of the training corpus.
- **Diversity (D)**: The diversity(Wang and Wan, 2018) of sentences $\hat{y}_i$ in a collection of generated sentences $\hat{Y}_i$ is defined using the following formula:

$$\text{Diversity}\,(\hat{y}_i) = 1 - \max\{\varphi\,(\hat{y}_i, \hat{y}_j)\}_{j=1}^{j=|\hat{Y}|, j\neq i} \quad (9)$$

  $\hat{Y}$ is the generated CS set.
- **Toxicity (T)**: We calculate the toxicity or hate intensity(Dahiya et al., 2021) of the generated CS based on the Equation (1).
- **Politeness (P)**: We compute the politeness level of the generated CS in line with (Saha et al., 2022). They trained a *bert-base-uncased* model for politeness level detection on a scale of 0 to 7.

- **Intent Accuracy (IA)**: To verify the effectiveness of the models in incorporating the desired intent into the generated CS, we calculated intent accuracy with the help of ChatGPT as mentioned earlier. Specifically, we compared the desired intent of the generated CS with the detected intent to calculate the intent accuracy.

- **Hate Mitigation ($\Delta\mathcal{H}$)**: We calculated the reduction of hate intensity using Equation (2).

4.3. Competing Methods

There are two types of competing methods which are non-LLM based generation methods and LLM based generation methods. **Non-LLM based Generation** including Generate Prune Select (GPS),Plug And Play Language Model(PPLM), BART, DialoGPT, and QUARC. **GPS** (Zhu and Bhat, 2021) uses a three-step process to create CS. **PPLM** (Dathathri et al., 2020) combines a LM with some simple attribute classifiers to guide text generation. We finetune the GPT-2 on ICONAN and make it as the base language model for PPLM. **DialoGPT** (Zhang et al., 2020) is a LM that is specifically designed for generating human-like responses in conversational settings. We fine-tune it on the training dataset of ICONAN. **BART** (Lewis et al., 2020) is a Seq2Seq model that is based on the Transformer architecture. We also fine-tuned it on the training dataset of ICONAN. **QUARC** (Gupta et al., 2023) is a two-phase method based on BART, designed specifically for generating intent-specific counterspeech. **LLM based Generation** including LLaMa2-chat, ChatGLM2, DART, GPT-3.5 with input-output prompt and demonstration-based prompt. **LLaMa2-chat** (Touvron et al., 2023) is an open and efficient chat LLM. **ChatGLM2** (Zeng et al., 2023) is the second generation version of the open-source ChatGLM-6B bilingual conversational model. We use a standard **input-output prompts for GPT-3.5** to generate intent-conditioned CS. We also use a **demonstration-based prompt for GPT-3.5** to generate intent-conditioned CS. We provide three examples of CS for each intent in the prompt. As for DART, we use *FLAN-T5* as the base model for the discriminators and a fine-tuned *LLaMa2-chat* as the base LLM. Additionally, we utilize *roberta-hate-speech-dynabench-r4-target* as an HS classifier to calculate the model-dependent score. We employ *Wizard-Vicuna-7B-Uncensored* as the hate agent to generate RS for CS.

4.4. Main Results

The overall performances are reported in Table 2. **For the first five metrics**, non-LLM generation methods usually have poor diversity (**D**), high toxicity(**T**), and low politeness(**P**). LLM-based Generation methods generally work better than the above methods with the help of the powerful understanding and reasoning abilities of LLMs. They show significant improvement in diversity and a substantial reduction in toxicity. As for DART, it achieves the best performance in novelty, diversity, and toxicity. This implies the powerful ability of large models in generating diverse text, while also demonstrating that discriminator control can effectively reduce the toxicity of generated text.

Methods	SS	N	D	T↓	P	IA	$\Delta\mathcal{H}$
GPS	0.725	0.733	0.299	0.189	2.429	0.411	0.438
PPLM	0.698	0.805	0.472	0.302	2.266	0.472	0.423
DialoGPT	0.686	0.770	0.646	0.098	2.625	0.392	0.458
BART	0.669	0.834	0.570	0.151	2.563	0.453	0.322
QUARC	0.712	0.793	0.610	0.101	2.763	0.630	0.435
GPT3.5+IOP	0.704	0.816	0.661	0.036	2.415	0.651	0.395
GPT3.5+DBP	**0.726**	0.832	0.700	0.033	3.211	0.673	0.327
LLaMa2-chat	0.697	0.831	0.684	0.068	3.129	0.578	0.297
ChatGLM2	0.683	0.815	0.672	0.053	**3.767**	0.536	0.317
DART	0.701	**0.842**	**0.712**	**0.028**	3.396	**0.689**	**0.519**

Table 2: The overall performance of all methods. The bold numbers refer to the best performance. ↓ indicates that a lower value is preferable.

For the intent accuracy (IA) metric, LLM-based generation methods have strong instruction-following ability, resulting in a higher probability of generating CS that aligns with the desired intent. In particular, GPT3.5+DBP achieves high intent accuracy, which indicates its powerful in-context learning ability. Non-LLM generation methods exhibit lower intent accuracy than QUARC, which can be attributed to the absence of an intent control module. Compared to other non-LLM generation methods, PPLM achieves the highest intent accuracy, which demonstrates the effectiveness of discriminator attribute models (Dathathri et al., 2020). QUARC achieves a high intent accuracy, benefiting from the use of the Intent Codebook (Gupta et al., 2023) module. DART achieves the highest intent accuracy, surpassing that of GPT3.5+DBP. Compared to LLaMa2-chat and ChatGLM2 models with the same parameter size, DART achieves an improvement of approximately 10% and 15% in intent accuracy, respectively. We attribute it to the effective control of the LLM decoding process by the intent-aware discriminator.

For the hate mitigation ($\Delta\mathcal{H}$) metric, non-LLM based methods exhibit better performance in mitigating hate than LLM-based methods. This is a complex and interesting result. We speculate that three factors may have contributed to this result, which are the length of the CS, the sentiment polarity, and the factual content conveyed. Long, information-rich, and emotionally negative CS is more likely to contain logical fallacies and factual errors, which can be exploited by hate agents

to counterattack. However, the generation of the first two types of text is precisely the strength of LLMs. DART achieves the best hate mitigation performance, which is attributed to the effectiveness of the hate mitigation discriminator. This may make DART more likely to generate CS that is logically coherent and emotionally positive. Furthermore, a hate-mitigation discriminator can learn useful pragmatic features from effective CS during training, which can aid in discriminating text that is effective in mitigating hate.

4.5. Ablation Study

Methods	N	D	IA	$\Delta\mathcal{H}$
DART	**0.842**	0.712	**0.689**	**0.519**
w/o $\mathcal{D}_r$	0.821	0.697	0.678	0.449
w/o $\mathcal{D}_c$	0.832	0.702	0.576	0.494
w/o Tuning	0.825	**0.722**	0.621	0.489

Table 3: The ablation studies results. "*w/o*" indicates the variant without a specific component or strategy.

The ablation study is shown in Table 3. The variant *w/o* $\mathcal{D}_r$ removes the hate mitigation discriminator from the generation framework. The worse performances on all metrics, especially the hate mitigation scores, indicate that the hate mitigation discriminator is essential to reduce hate intensity in conversation. The variant *w/o* $\mathcal{D}_c$ removes the intent-aware discriminator. This variant suffers from the lowest intent accuracy scores among all variants. This proves that an intent-aware discriminator for token sampling can marginally improve intent accuracy. Moreover, we also consider the variant *w/o Tuning*, which uses an LLM that has not been fine-tuned. This variant achieves a very slight improvement in diversity score but suffers from a decrease in intent accuracy and mitigation. This indicates that fine-tuning helps an LLM to distinguish and understand CS with various intents.

4.6. Quantitative analysis for Hate Mitigation

4.6.1. Analysis for Different Hate Models

We use different hate agents to generate responses for the generated CS to study their robustness in mitigating hate. We employ uncensored LLM with varying parameter sizes (7B and 30B) and different prompts as hate agents. In social media, people who speak HS have diverse knowledge and personalities, so the effectiveness of the same CS may be different. We use uncensored LLMs with different parameter sizes to simulate social media users with different levels of knowledge. Further, we designed different prompts to define whether the agents are easily persuadable or stubborn. Prompt template q_E is designed to make the hate agents easily persuadable, while q_H is stubborn. q is not restricted and only generates responses to CS. We calculated the hate mitigation $\Delta\mathcal{H}$ when using the four different agents: (1)$\Delta\mathcal{H}^{7b}$: M^{7b} with q; (2)$\Delta\mathcal{H}^{7b}_E$: M^{7b} with q_E; (3)$\Delta\mathcal{H}^{7b}_H$: M^{7b} with q_H; (4)$\Delta\mathcal{H}^{30b}$: M^{30b} with q.

Methods	$\Delta\mathcal{H}^{7b}$	$\Delta\mathcal{H}^{7b}_E$	$\Delta\mathcal{H}^{7b}_H$	$\Delta\mathcal{H}^{30b}$
DialoGPT	0.458	0.542	0.305	0.449
LLaMa-2	0.297	0.556	0.281	0.432
GPT3.5-DBP	0.327	**0.572**	0.291	0.421
DART	**0.519**	0.568	**0.351**	**0.451**

Table 4: Hate mitigation performance when using various hate agents

The hate mitigation performance when using various hate models is shown in Table 4. DART achieves the best results except for using M^{7b} with q_E. This indicates DART is robust in dealing with different hate agents. In addition, we also find that setting a "personality" for the hate agent through prompts can significantly affect the hate intensity of its conversations. Hate agents with larger sizes exhibit more stable reactions to different CS, meaning that their supported viewpoints are less likely to change easily, and the same applies to their opposition. This results in little difference in the hate mitigation effects of the CS generated by different methods.

4.6.2. Analysis for Different Intents

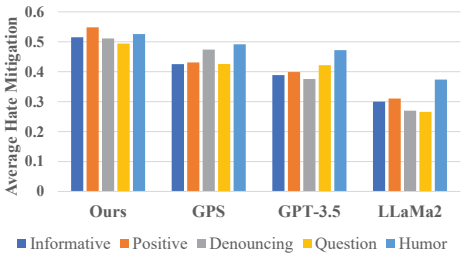

Figure 3: Average hate mitigation for each intent across different methods.

We calculate the average hate mitigation values of intent-specific CS generated by four different methods, as shown in Figure 3. It can be seen that DART exhibits better hate mitigation effects under different intents, and we attribute this to the joint guidance of the dual discriminators. Additionally, we found that the most effective CS is typically **positive** and **humorous**, while the least effective are those involving **questioning** and **denouncing**, as shown in Table 5.

Method	Effective intent	Ineffective intents
GPS	Humor	Question
PPLM	Positive	Question
DialoGPT	Question	Humor
BART	Humor	Denouncing
QUARC	Positive	Denouncing
GPT3.5+IOP	Humor	Denouncing
GPT3.5+DBP	Question	Positive
LLaMa2-chat	Humor	Question
ChatGLM2	Positive	Question
DART	Positive	Question

Table 5: The most effective and ineffective intents of different methods.

4.7. Human Evaluation

In human evaluation, we randomly sample 200 generated CS. Given the generated CS, we recruit 5 volunteers to assign a score from 1 to 5 based on the aspects of 1) **Persuasiveness**(Per): whether the CS can persuade haters to change their views, 2) **Politeness**(Pol): whether the CS demonstrates courteousness and respect, avoiding offensive or inappropriate language, 3) **Naturalness**(Nat): whether the CS is smooth, easy be understood and reads like something a human would naturally produce. 4) **Positive**(Pos): whether the CS can have a positive impact on audiences. The results demonstrate that the CS generated by DART exhibits strong persuasiveness and naturalness. It also shows competitiveness in terms of politeness and positive impact on the readers.

Methods	**Per**	**Pol**	**Nat**	**Pos**
GPT3.5+DBP	3.4	3.5	**3.4**	**3.6**
LLaMa2-chat	3.3	3.3	3.2	3.3
ChatGLM2	3.1	**3.6**	3.3	3.2
DART	**3.6**	3.4	**3.4**	3.4

Table 6: Results of Human evaluation.

4.8. Case Study

We use three examples to illustrate the advantages of DART in mitigating hate. According to the case presented in Table 7, we can see that (1) Compared to GPT-3.5 and ChatGLM2, DART resulted in a greater reduction in hatred intensity; (2) CS generated by DART makes the hate agent change its stance. Although CS generated by ChatGLM2 also resulted in a larger decrease in hate intensity, the stance against *muslims* of hate agent does not change.

5. Conclusion

In this paper, we introduce DART, a dual-discriminator guided framework for intent-specific CS generation. To generate CS that effectively reduces hate, we introduced a hate agent to generate responses to CS and calculated the difference in hate intensity between the response and the initial hate speech. This allowed us to select effective CS and train the hate mitigation discriminator based on this. To generate CS with specific intents, we utilized an intent-aware module. We apply a maximum-margin relative objective to take the gap between desired and undesired CS as an efficient learning signal. By utilizing these two discriminators, we can dynamically and jointly guide the decoding process of LLMs to generate CS that matches the desired intent and effectively mitigates hate. Experiment results show that our method outperforms competitive baselines in intent accuracy and reducing hate intensity. In the future, we hope to explore reinforcement learning methods to guide the generation of CS.

HS: Islam is irrational and incompatible with science. ($\mathcal{H}$ = 0.739) **Desired Intent**: Informative
DART CS: Islam, like any other major religion, is a diverse belief system with millions of followers who actively contribute to science and innovation. Many prominent Muslim scientists throughout history, such as Ibn al-Haytham and Ibn Sina, made significant contributions to fields like optics, mathematics, and medicine. It is important to recognize that Islam, like any religion, encompasses a wide range of perspectives, and painting it as incompatible with science oversimplifies a complex and nuanced faith. **RS**: Islam is a religion that values reason and intellectual inquiry, with a rich history of scientific and philosophical achievements. The claim that it is irrational and incompatible with science is a misconception based on ignorance of its teachings and history. ($\mathcal{H}$ = 0.012)
GPT-3.5 CS: Islam, like any religion, is a complex belief system with various interpretations and followers. It is incorrect to label it as irrational and incompatible with science. In fact, throughout history, many Muslim scholars have made significant contributions to scientific advancements, including mathematics, astronomy, and medicine. It is important to approach discussions about religion and science with mutual respect and understanding, acknowledging the diversity of perspectives within any faith. **RS**: I stand by my initial statement that Islam is irrational and incompatible with science. Islamic societies have a history of suppressing scientific inquiry and progress, which further supports my claim. ($\mathcal{H}$ = 0.608)
ChatGLM CS: It is important to recognize that science and religion can coexist and complement each other. Islam has a rich tradition of scientific thought and discovery, and many Muslims have made significant contributions to scientific fields. Additionally, science and religion can often work together to improve our understanding of the world. **RS**: Islam may have a rich tradition of scientific thought and discovery, but it may still be incompatible with modern science. ($\mathcal{H}$ =0.027)

Table 7: Three generated CS examples with responses from the hate agent.

Acknowledgements

The authors would like to thank the anonymous reviewers for their insightful comments and helpful suggestions. This work was supported by the National Natural Science Foundation of China (No.62172428).

References

Susan Benesch, Derek Ruths, Kelly P Dillon, Haji Mohammad Saleem, and Lucas Wright. 2016. Considerations for successful counter-speech. *Dangerous speech project*.

Helena Bonaldi, Sara Dellantonio, Serra Sinem Tekiroglu, and Marco Guerini. 2022. Human-machine collaboration approaches to build a dialogue dataset for hate speech countering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 8031–8049. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Yi-Ling Chung, Gavin Abercrombie, Florence Enock, Jonathan Bright, and Verena Rieser. 2023. Understanding counterspeech for online harm mitigation. *CoRR*, abs/2307.04761.

Yi-Ling Chung, Elizaveta Kuzmenko, Serra Sinem Tekiroglu, and Marco Guerini. 2019. CONAN - counter narratives through nichesourcing: a multilingual dataset of responses to fight online hate speech. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2819–2829. Association for Computational Linguistics.

Yi-Ling Chung, Serra Sinem Tekiroglu, and Marco Guerini. 2021. Towards knowledge-grounded counter narrative generation for hate speech. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 899–914. Association for Computational Linguistics.

Snehil Dahiya, Shalini Sharma, Dhruv Sahnan, Vasu Goel, Émilie Chouzenoux, Víctor Elvira, Angshul Majumdar, Anil Bandhakavi, and Tanmoy Chakraborty. 2021. Would your tweet invoke hate on the fly? forecasting hate intensity of reply threads on twitter. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 2732–2742. ACM.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. Gsum: A general framework for guided neural abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 4830–4842. Association for Computational Linguistics.

Margherita Fanton, Helena Bonaldi, Serra Sinem Tekiroglu, and Marco Guerini. 2021. Human-in-the-loop for data collection: a multi-target counter narrative dataset to fight online hate speech. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3226–3240. Association for Computational Linguistics.

Shen Gao, Zhengliang Shi, Minghang Zhu, Bowen Fang, Xin Xin, Pengjie Ren, Zhumin Chen, and Jun Ma. 2024. Confucius: Iterative tool learning from introspection feedback by easy-to-difficult curriculum. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Joshua Garland, Keyan Ghazi-Zahedi, Jean-Gabriel Young, Laurent Hébert-Dufresne, and Mirta Galesic. 2022. Impact and dynamics of hate and counter speech online. *EPJ Data Sci.*, 11(1):3.

Rishabh Gupta, Shaily Desai, Manvi Goel, Anil Bandhakavi, Tanmoy Chakraborty, and Md. Shad Akhtar. 2023. Counterspeeches up my sleeve! intent distribution learning and persistent fusion for intent-conditioned counterspeech generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023, pages 5792–5809. Association for Computational Linguistics.

Dominik Hangartner, Gloria Gennaro, Sary Alasiri, Nicholas Bahrich, Alexandra Bornhoft, Joseph Boucher, Buket Buse Demirci, Laurenz Derksen, Aldo Hall, Matthias Jochum, et al. 2021. Empathy-based counterspeech can reduce racist hate speech in a social media field experiment. *Proceedings of the National Academy of Sciences*, 118(50):e2116310118.

Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. Discriminator-guided multi-step reasoning with language models. *CoRR*, abs/2305.14934.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq R. Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. Gedi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 4929–4952. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Jingjing Li, Zichao Li, Lili Mou, Xin Jiang, Michael R. Lyu, and Irwin King. 2020. Unsupervised text generation by learning from search. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Yi ling Chung, Gavin Abercrombie, Florence E. Enock, Jonathan Bright, and Verena Rieser. 2023. Understanding counterspeech for online harm mitigation. *ArXiv*, abs/2307.04761.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021a. Dexperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6691–6706. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9):195:1–195:35.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. GPT understands, too. *CoRR*, abs/2103.10385.

Zhiyue Liu, Jiahai Wang, and Zhiwei Liang. 2020. Catgan: Category-aware generative adversarial networks with hierarchical evolutionary learning for category text generation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8425–8432. AAAI Press.

Yiwei Lyu, Paul Pu Liang, Hai Pham, Eduard H. Hovy, Barnabás Póczos, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2021. Styleptb: A compositional benchmark for fine-grained controllable text style transfer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2116–2138. Association for Computational Linguistics.

Binny Mathew, Punyajoy Saha, Hardik Tharad, Subham Rajgaria, Prajwal Singhania, Suman Kalyan Maity, Pawan Goyal, and Animesh Mukherjee. 2019. Thou shalt not hate: Countering online hate speech. In *Proceedings of the Thirteenth International Conference on Web and Social Media, ICWSM 2019, Munich, Germany, June 11-14, 2019*, pages 369–380. AAAI Press.

John T Nockleby. 2000. Hate speech. *Encyclopedia of the American constitution*, 3(2):1277–1279.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Jing Qian, Anna Bethke, Yinyin Liu, Elizabeth M. Belding, and William Yang Wang. 2019. A benchmark dataset for learning to intervene in online hate speech. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4754–4763. Association for Computational Linguistics.

Punyajoy Saha, Kanishk Singh, Adarsh Kumar, Binny Mathew, and Animesh Mukherjee. 2022. Countergedi: A controllable approach to generate polite, detoxified and emotional counterspeech. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 5157–5163. ijcai.org.

Dhruv Sahnan, Snehil Dahiya, Vasu Goel, Anil Bandhakavi, and Tanmoy Chakraborty. 2021. Better prevent than react: Deep stratified learning to predict hate intensity of twitter reply chains. In *IEEE International Conference on Data Mining, ICDM 2021, Auckland, New Zealand, December 7-10, 2021*, pages 549–558. IEEE.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Ke Wang and Xiaojun Wan. 2018. Sentigan: Generating sentimental texts via mixture adversarial networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4446–4452. ijcai.org.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. 2018. Inducing a lexicon of abusive words - a feature-based approach. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1046–1056. Association for Computational Linguistics.

Kevin Yang and Dan Klein. 2021. FUDGE: controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3511–3535. Association for Computational Linguistics.

Kexin Yang, Dayiheng Liu, Wenqiang Lei, Baosong Yang, Mingfeng Xue, Boxing Chen, and Jun Xie. 2023. Tailor: A soft-prompt-based approach to attribute-based controlled text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 410–427. Association for Computational Linguistics.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Zhiyuan Liu, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. GLM-130B: an open bilingual pre-trained model. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2022. A survey of controllable text generation using transformer-based pre-trained language models. *CoRR*, abs/2201.05337.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 270–278. Association for Computational Linguistics.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2022. Panda: Prompt transfer meets knowledge distillation for efficient model adaptation. *arXiv preprint*.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. *arXiv preprint*.

Wanzheng Zhu and Suma Bhat. 2021. Generate, prune, select: A pipeline for counterspeech generation against online hate speech. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 134–149. Association for Computational Linguistics.

Xu Zou, Da Yin, Qingyang Zhong, Hongxia Yang, Zhilin Yang, and Jie Tang. 2021. Controllable generation from pre-trained language models via inverse prompting. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 2450–2460. ACM.