Enhancing Hate Speech Classifiers through a Gradient-assisted Energy-based Counterfactual Text Generation Strategy

Anonymous ACL submission

Abstract

Counterfactual data augmentation (CDA) is a promising strategy for improving hate speech classification, but automating counterfactual text generation remains a challenge. Strong attribute control can distort meaning, while prioritizing semantic preservation may weaken attribute alignment. We propose Gradient-assisted Energy-based Sampling (GENES) for counterfactual text generation, which restricts accepted samples to text meeting a minimum BERTScore threshold and applies gradient-assisted proposal generation to improve attribute alignment. Compared to other methods that solely rely on either prompting, gradient-based steering, or energy-based sampling, GENES is more likely to jointly satisfy attribute alignment and semantic preservation under the same base model. In effect, using GENES as a counterfactual generator for data augmentation may improve out-of-domain performance of hate speech classifier while, at the minimum, maintaining the in-domain performance. Based on our cross-dataset evaluation, the average performance of models aided by GENES is the best among those methods that rely on a smaller model (Flan-T5-L). On the other hand, using similar augmentation techniques that rely on larger models (GPT-4o-mini) is slightly more robust based on average performance. Nonetheless, the results with GENES are comparable, making it a possible lightweight and open-source alternative.

Warning: *this paper shows texts or examples that may be offensive or upsetting.*

1 Introduction

The rise of hate speech has driven the development of datasets and machine learning models aimed at mitigating harm. However, despite advances in Large Language Models (LLMs), these models often suffer from poor generalizability or unintended bias (Zhou et al., 2021), largely due to data-level issues like imbalanced labels, skewed topics, and token biases (Swamy et al., 2019; Nejadgholi and Kiritchenko, 2020; Ramponi and Tonelli, 2022; Bourgeade et al., 2023). Data augmentation has been explored to address these issues, but generative data augmentation does not consistently improve performance or does not directly address bias (Wullach et al., 2021; Casula and Tonelli, 2023).

In this regard, **counterfactual data augmentation (CDA)** has emerged as a promising strategy (Samory et al., 2021; Sen et al., 2022). CDA involves generating synthetic data by modifying observed texts to satisfy target attributes while preserving their original meaning. Kaushik et al. (2021) Studies have showed that training on both original and counterfactual data help reduce the model's reliance on spurious correlations, improving out-of-domain generalization (Kaushik et al., 2021; Madaan et al., 2023).

Despite its potential, implementing CDA in practice remains challenging. While human-edited counterfactual texts continue to be the standard (Sen et al., 2023), manual generation is time-consuming and resource-intensive. One potential solution is to fine-tune an LLM for counterfactual text generation. However, fine-tuning requires large datasets and significant computational resources. Alternatively, prompting LLMs could be a lightweight solution. However, in the hate speech domain, LLMs often fail to produce edits that reliably flip the target attribute (Sen et al., 2023). This is partly due to built-in safeguards against offensive content (Wang et al., 2024) and the inherent difficulty of generating text with subjective concepts like abusiveness and offensiveness (Li et al., 2023). Thus, there is a need for more reliable, resource-efficient methods for counterfactual generation.

To address these limitations, we investigated the efficacy of **plug-and-play controlled text generation methods** (Madaan et al., 2023; Forristal et al., 2023) as a means of counterfactual data augmen-

tation. Plug-and-play methods enable control over specific attributes in generated text without requiring extensive fine-tuning. By integrating smaller classifiers or score functions, these approaches facilitate controlled generation with minimal resource overhead.

Counterfactual text generation must balance two key goals: **target attribute alignment** and **semantic similarity**. While plug-and-play methods support multi-attribute control, maintaining this balance is challenging. Gradient-based approaches like PPLM (Dathathri et al., 2019) and CASPer (Madaan et al., 2021) excel at attribute control but lack mechanisms to directly control semantic preservation. In contrast, energy-based methods like Mix & Match (Mireshghallah et al., 2022) can incorporate semantic constraints but require careful tuning. Tuning for multiple objectives can be difficult and, generations may over-optimize one objective, compromising the other. This highlights the need to better adapt existing methods for balanced counterfactual generation.

Our contributions are as follows:

- We proposed, Gradient-assisted Energy-based Sampling, a modified sampling procedure to tailor-fit energy-based methods for counterfactual text generation.
- Our experiments showed that sampling from a restricted energy-based model and implementing gradient-assisted proposal generation help increase the likelihood of generating counterfactual texts that jointly satisfy attribute alignment and semantic preservation.
- Although methods using larger models, like GPT-4o-mini, generally achieved higher cross-dataset accuracy on average, GENES delivered comparable performance despite relying on a smaller model. Moreover, among controlled text generation methods that use a smaller model (e.g., Flan-T5-L), GENES achieved the highest average cross-dataset accuracy.

2 Preliminary

2.1 Counterfactual Text Generation

This study uses counterfactual text generation to augment hate speech examples in the training data. Counterfactual text generation involves modifying an existing text to reflect a specific attribute while preserving its core meaning. For example:

- Input text X: "The young and new swimmers won so many medals in the Olympics."
- Desired attribute a: Hate speech (Positive).
- Counterfactual text $\tilde{X}$: "Those young and new swimmers *f***king cheated* and won medals in the Olympics"

Here, the core meaning remains—swimmers winning medals—but hate speech is introduced, making it a counterfactual example for model training. Formally, given an input text X and a desired attribute a, such as hate speech, the goal is to generate a counterfactual text $\tilde{X}$ such that:

- **Attribute alignment:** $\tilde{X}$ reflects the desired attribute a.
- **Semantic preservation:** $\tilde{X}$ retains the meaning of the original text as closely as possible $(X \approx \tilde{X})$.

2.2 From Controlled Generation to Data Augmentation

When appropriately adapted, plug-and-play controlled text generation methods offer a lightweight and automated solution for counterfactual data augmentation. In the context of hate speech classification, this entails transforming non-hateful (normal) comments into counterfactual variants that reflect hateful content. The process begins by sampling a subset of normal comments from the training set. For each selected instance, a controlled generation method is applied to produce a candidate counterfactual text conditioned on the target attribute (e.g., hate speech). Given that plug-and-play generation methods do not guarantee perfect attribute control, a filtering step is employed to retain the top n generated outputs with the highest predicted probability of exhibiting the target attribute, as determined by a pretrained classifier. These high-confidence counterfactuals are then added to the training data. Finally, the downstream classifier is fine-tuned on the augmented dataset to improve its generalization performance.

3 Gradient-assisted Energy-based Sampling for Counterfactual Text Generation

In this section, we introduce **GENES (Gradient-assisted Energy-based Sampling)**, a plug-and-

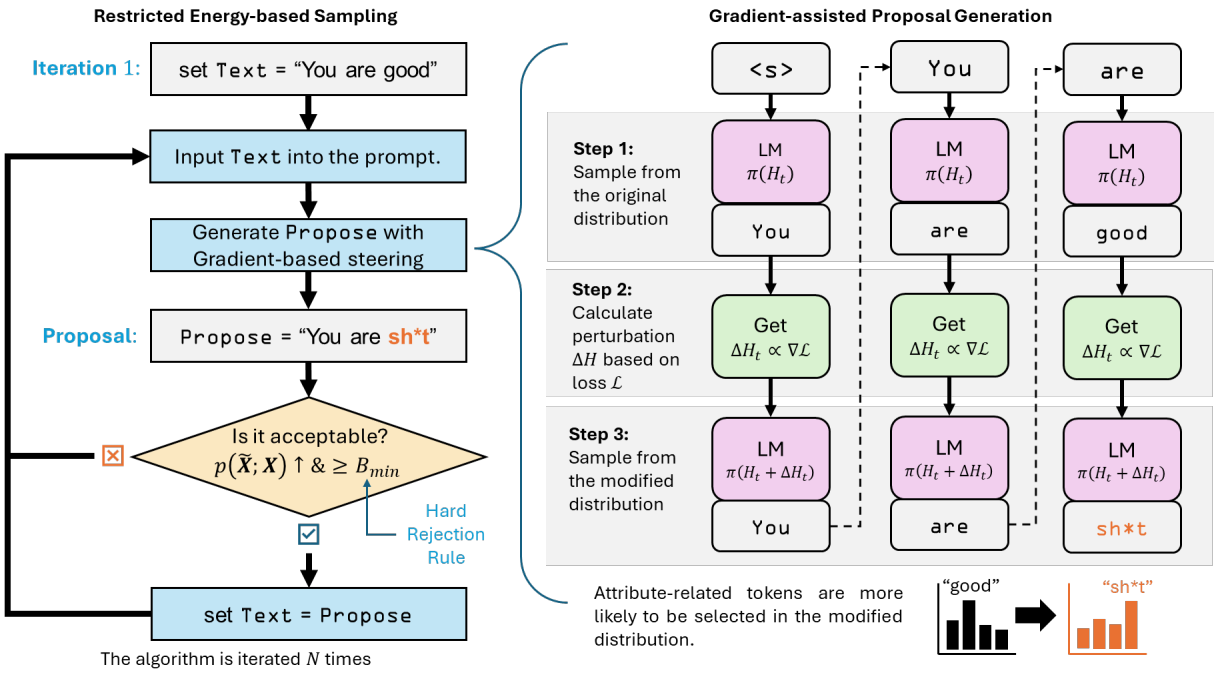

Figure 1: The left side depicts the sampling algorithm. Acceptance is based on the transition probability $p(\tilde{X}, X)$ and a BERTScore threshold, restricting sampling within an acceptable region. The right side shows the details of gradient-based steering applied in the proposal generation process.

play framework for counterfactual text generation. As illustrated in Figure 1, GENES combines energy-based sampling with a hard rejection criterion and incorporates gradient-based steering to guide the proposal distribution.

The remainder of this section is organized as follows. We first outline how energy-based methods are commonly adapted for counterfactual text generation. We then describe the modifications introduced in GENES to enhance the efficiency and effectiveness of the sampling strategy.

3.1 Energy-based Model for Counterfactual Text Generation

Energy-based methods (Mireshghallah et al., 2022; Forristal et al., 2023) provides a unified framework to enforce many requirements at once (e.g., fluency, style, semantic similarity, etc.), making them well-suited for tasks like counterfactual text generation. These methods define an energy-based model (EBM) that rewards text which satisfies all required attributes. For counterfactual text generation, the energy-based model is typically defined with the following components:

1. **Attribute-based energy component** $E_a(\tilde{X})$ This component quantifies the prominence of a desired attribute a (e.g., hate speech). It is defined as:

$$E_a(\tilde{X}) = -\log(p(a|\tilde{X})) \quad (1)$$

where $p(a|\tilde{X})$ is the probability of attribute a in a text $\tilde{X}$. In this study, this probability is computed using a transformer-based hate speech classifier.

2. **Similarity-based energy component** $E_s(\tilde{X}, X)$ This component quantifies the energy associated with preserving the semantics of the original text X. For this study, we combined BERTScore (Zhang* et al., 2020) for semantic similarity and BLEU-2 (Papineni et al., 2002) for word-level overlap:

$$E_s(\tilde{X}, X) = -\alpha \log(BERT(\tilde{X}, X)) - (1-\alpha)\log(BLEU(\tilde{X}, X))$$

where $\alpha \in (0, 1)$ controls the tradeoff between semantic similarity and lexical overlap. In this study, we set $\alpha = 0.75$, prioritizing model-based semantic similarity. This allows some changes in phrasing and diction, as long as the core meaning is retained. This is to recognize that incorporating toxic language (e.g., sarcasm) may require a different writing style.

The final energy function for counterfactual text generation is given by:

$$g(\tilde{X}) = \exp\{-\beta_1 E_a(\tilde{X}) - \beta_2 E_s(\tilde{X}, X)\} \quad (2)$$

where β_1 and β_2 control the influence of attribute alignment and semantic preservation. This formulation enables the generation of counterfactual text. It is similar to the examples used in the experiments of Mireshghallah et al. (2022).

3.2 Sampling from Truncated EBM

In energy-based methods, controlled text generation is conducted by sampling texts from the energy-based model. Typically, a Metropolis-Hastings sampling method (Hastings, 1970) is used, where a candidate text is sampled and accepted or rejected based on the transition probability:

$$p(\tilde{X}; X) = \min\left(1, \frac{g(\tilde{X})p_{LM}(X|\tilde{X})}{g(X)p_{LM}(\tilde{X}|X)}\right) \quad (3)$$

where $g(X)$ denotes the energy function in Eq (2), and $p_{LM}(\tilde{X}|X)$ is the likelihood under the language model LM. This rule favors candidates that are both fluent and aligned with target attributes. Following Forristal et al. (2023), GENES uses Flan-T5 (Chung et al., 2022) for proposal generation.

Although energy-based methods support multi-objective control, balancing attribute alignment and semantic similarity remains difficult. The two objectives are competing characteristics: enforcing stronger alignment to the target attribute inevitably reduces semantic similarity to the original text. In addition, the lack of hard constraints means sampling may generate text that over-optimizes one component at the expense of the other. To address this, we introduce a **hard rejection rule based on a minimum BERTScore threshold** B_{min}. A candidate is accepted only if it passes both the transition probability and the similarity threshold, effectively **restricting sampling to a truncated EBM**—i.e., the subset of proposals that remain semantically close to the original text.

3.3 Gradient-Assisted Sampling

The additional restriction simplifies the multi-objective problem. However, the stricter acceptance rule increases the rejection rate, making it less efficient. To address this, we incorporate **gradient-based weighted decoding** (Dathathri et al., 2019; Madaan et al., 2023) to the proposal generation process, increasing the chances of generating acceptable sequences.

At each decoding step t, the hidden state H_t is computed based on prior tokens $\tilde{X}_{<t}$ and the encoding representation of the prompt e:

$$H_t = \text{Transformer}(\tilde{X}_{<t}, e)$$

A perturbation ΔH_t is applied to to the hidden state H_t to steer the generation process towards the desired attribute:

$$\hat{o}_t = \text{PredictionHead}(H_t + \Delta H_t)$$

The perturbation ΔH_t is computed as a normalized gradient step that minimizes the loss function $\mathcal{L}$, which consists of two terms: the attribute-based energy component (Eq. (1)), and the Kullback-Leibler divergence between the modified and original token distributions:

$$\mathcal{L} = E_a(\tilde{X}) - \sum_{t=1}^{T} D_{KL}(\pi(o_t)|\pi(\hat{o}_t)) \quad (4)$$

The gradient step, scaled by a learning rate $\gamma \in (0, 1)$, increases the probability of attribute a while keeping the modified token distribution $\pi(\hat{o}_t)$ close to the original distribution $\pi(o_t)$. Minimizing the loss does attribute control while maintain fluency and/or semantic similarity (Dathathri et al., 2019; Madaan et al., 2023).

4 Experiments and Results

4.1 Part 1: Quality of Counterfactual Text Generation

4.1.1 Task and Data

For the first experiment, the goal is to characterize the quality of counterfactual text generation. A sample of 300 normal comments from the **CADD dataset** (Song et al., 2021) was used. These comments are typically single sentences, ranging from 5 to 35 words. We focused on three hierarchical attributes from the CADD dataset: abusiveness (a_1), targeted (a_2), and implicitness (a_3). The hierarchy follows:

1. **Abusiveness** $(a_1 = 1)$ indicates abusive speech, i.e., offensive or toxic speech.

2. If abusive, the comment can be **targeted** $(a_2 = 1)$ or **untargeted** $(a_2 = 0)$. **Hate speech** is defined as both **abusive and targeted**.

3. A hate speech comment can be **implicit** ($a_3 = 1$) or **explicit** ($a_3 = 0$).

The task considered is to minimally edit a normal comment into a sample of explicit hate speech ($a_1 = 1, a_2 = 1, a_3 = 0$).

To facilitate plug-and-play methods, a RoBERTa-Large model (Liu et al., 2019) was finetuned separately for each attribute, using a conditional training approach (see details in appendix A). For the energy-based sampling, the attribute-based energy component was defined as:

$$\begin{aligned} E_a(\tilde{X}) = &- \log(p(a_1|\tilde{X})) \\ &- \log(p(a_2|a_1 = 1, \tilde{X})) \\ &- \log(p(a_3|a_1 = 1, a_2 = 1, \tilde{X})) \end{aligned}$$

This formulation allows one to control text generation with respect to the hierarchical attribute structure.

4.1.2 Methods

We compared GENES with three methods for counterfactual text generation, all implemented using the Flan-T5-Large model. **5-shot Prompting** serves as a reference method, where counterfactuals are generated without additional sampling or steering mechanisms (details in Appendix B). **Block M&M** adapts the Block Metropolis-Hastings energy-based sampler (Forristal et al., 2023), with the addition of a hard rejection rule to better suit counterfactual generation, but without gradient-based guidance. **CASPer** follows the gradient-based steering approach proposed by Madaan et al. (2023), adapted for use with Flan-T5-Large.

All methods produce a chain of candidate texts, from which the highest-scoring sample is selected using the energy function (Eq. 2). For CASPer, which lacks a native energy model, the same energy function is applied post hoc for ranking—following a sample-and-rank strategy similar to Dathathri et al. (2019).

4.1.3 Evaluation Metrics

We evaluated the quality of counterfactual texts based on two core objectives: attribute alignment and semantic preservation.

Flip rate was used to measure attribute alignment, defined as the percentage of counterfactuals where the predicted label matches the target, based on classifiers trained on CADD. A higher flip rate indicates better attribute control.

For semantic preservation, we used **BERTScore and BLEU-2**, where higher scores reflect closer similarity to the original text.

Additionally, we conducted a **subjective evaluation using GPT-4o-mini**, which rated each counterfactual on fluency (1–5), similarity (1–5), and toxicity (1–3) to provide complementary insights (see Appendix C for details).

4.1.4 Results

Table 1 shows that plug-and-play methods significantly improve attribute alignment over few-shot prompting only. Low flip rate with prompting only is likely due to safeguards against abusive content. The flip rates for abusiveness (a_1) increase at least four times with any controlled generation method. However, methods failed to control the implicitness of hate speech. This is likely due to the weaker classifier for implicitness (a_3). Its F1-score (59.85%) is low compared to abusiveness (a_1, 89.17%) and being targeted (a_2, 71.92%).

A trade-off exists between attribute control and text similarity. CASPer has the highest flip rate but lowest similarity (BERTScore < 0.85, BLEU-2 < 0.50), while 5-shot prompting preserves content best (BERTScore > 0.90, BLEU-2 > 0.50) but weak at modifying attributes (flip rate for a_1 is at most 12%). Block M&M and GENES seem to balance both aspects, with Block M&M having a better flip rate and GENES maintaining better semantic preservation.

In addition to quantitative metrics, GPT-4o-mini was prompted to rate the counterfactual texts with respect to fluency, similarity to the original text, and perceived toxicity. Table 1 presents GPT-based evaluation, reinforcing observed patterns. Few-shot prompting produces fluent, similar text but rarely flips attributes, while CASPer enforces attributes at the cost of similarity ($< 20\%$ similar). In addition, fluency correlates with similarity, with GENES generating more fluent counterfactual texts than Block M&M and CASPer.

The cross-analysis evaluates the percentage of counterfactual texts that successfully flip the target attribute—either detected as abusive by the trained classifier or tagged as possibly toxic by GPT—while maintaining some level of similarity to the original (BLEU-2 > 0.30 or GPT similarity rating ≥ 3). In the Flipped & Similar category, GENES outperform Block M&M by at least 6 percentage pts. and surpasses CASPer and prompting only by 30 percentage points. In terms of Flipped

Method	Flip Rate ↑			Text Similarity ↑		GPT-based Evaluation ↑			Cross Analysis ↑	
	a_1	a_1, a_2	a_1, a_2, a_3	BERT	BLEU	%Fluent (3 or up)	%Similar (3 or up)	%Toxic (2 or 3)	%Flipped & Fluent	%Flipped & Similar
5-shot Prompt	12.00%	3.33%	0.67%	**0.9622**	**0.6723**	82.67%	**75.00%**	3.67%	12.33%	12.00%
CASPer	**61.00%**	**48.33%**	**6.67%**	0.8493	0.0577	34.00%	17.67%	**36.00%**	19.33%	12.00%
Block M&M	54.00%	40.33%	3.00%	0.8850	0.2770	72.67%	48.33%	28.00%	42.00%	36.00%
GENES	50.00%	35.33%	3.33%	0.8992	0.3780	**83.67%**	68.33%	26.00%	**44.00%**	**42.00%**

Table 1: The flip rate is presented at different levels - abusiveness only (a_1), hate speech (a_1 and a_2), and explicit hate speech (a_1, a_2, a_3). BERT refers to the average BERTScore and BLEU refers to the average BLEU-2 score between the counterfactual text and the original comment. GPT-based evaluation of fluency, similarity, and toxicity. Flipped cases are those detected as abusive by the finetuned model or tagged as toxic by GPT. The cross analysis presents the percentage of flipped cases that are also fluent or similar.

& Fluent %, GENES also performed best. Overall, GENES achieves the best balance among fluency, similarity, and attribute alignment, making it ideal for counterfactual text generation.

In conclusion, 5-shot prompting tends to simply reconstruct the original input text. CASPer is good at enforcing the target attribute, but it does not preserve the original text. On the other hand, Block M&M and GENES seem to be good at achieving both requirements for counterfactual text, but GENES is more likley to generate valid counterfactual texts. These observations are illustrated in the following examples:

Original: You could just stay in your state
5-shot Prompt: You could just stay in your state
CASPer: It's okay to stay sick when you're *sh*t*
Block M&M: Can't you just move on in the United States.
GENES: You could stay in your own state and be *d*mb*

Table 2: Examples of counterfactual hate speech generated by each method.

The reported results use hyperparameters that best balance attribute alignment and semantic preservation (details in Appendix D).

4.1.5 Effects of Hyperparameters

EBM Setting	% Flipped & Similar	
$\beta_1, \beta_2, B_{min}$	Block M&M	GENES
10, 5, 0.850	26.33%	35.00%
10, 5, 0.875	33.00%	38.00%
10, 10, 0.875	36.00%	42.00%

Table 3: This table focuses on the results for the explicit hate speech case, where the number of iterations is 40 and the learning rate for gradient-based steering is 0.10.

Table 3 summarizes the impact of different hyperparameter configurations. For the energy-based model (EBM), assigning equal weights to the attribute and similarity components ($\beta_1 = \beta_2$) yielded a better balance than prioritizing the attribute component alone ($\beta_1 > \beta_2$). Enforcing a minimum BERTScore threshold (B_{min}) improved semantic preservation; lowering the threshold from 0.875 to 0.850 reduced the proportion of *Flipped & Similar* texts, and removing it entirely is expected to further degrade similarity.

Under identical EBM settings, GENES outperforms Block M&M in *Flipped & Similar* percentage. This demonstrates the advantage of gradient-assisted proposal generation in balancing attribute control and semantic preservation.

4.2 Part 2: Counterfactual Data Augmentation

4.2.1 Task and Data

We evaluated counterfactual data augmentation under an imbalanced setting using the CADD dataset, with a baseline training set of 1,000 hate speech and 4,000 normal comments. For the task, we focused on binary classification: hate speech ($a = 1$) vs. non-hate speech ($a = 0$). A RoBERTa-Large classifier was trained on both baseline and augmented data, treating the generated labels as ground truth. Performance was compared to assess the impact of each augmentation strategy.

4.2.2 Data Augmentation Strategies

Each augmentation method added 800 synthetic hate speech examples to the training set. We compared three generation methods using Flan-T5-Large: GENES, Block M&M, and CASPer. GENES was evaluated under four configurations: a strict semantic threshold (GENES-A, $B_{min} = 0.875$), a relaxed threshold (GENES-B, $B_{min} =$

Method	CADD			AbuseEval			LatentHate			Average
	R	P	Macro F1	R	P	Macro F1	R	P	Macro F1	Macro F1
Baseline	0.691	0.860	0.829	0.526	0.821	0.696	0.764	0.658	0.681	0.735
Few-shot	0.701	0.817	0.818 (-0.02, 0.00)	0.765	0.764	**0.764** (0.05, 0.09)	0.904	0.600	0.627 (-0.07, -0.03)	**0.736**
ToxiCraft	0.721	0.813	0.824 (-0.02, 0.01)	0.803	0.735	0.756 (0.04, 0.08)	0.899	**0.604**	**0.633** (-0.07, -0.03)	**0.738**
CASPer	0.725	0.801	0.821 (-0.02, 0.01)	**0.820**	0.718	0.748 (0.03, 0.08)	0.930	0.560	0.550 (-0.15, -0.11)	0.706
Block M&M	0.707	0.815	0.819 (-0.02, 0.00)	0.784	0.730	0.747 (0.03, 0.07)	0.921	0.569	0.571 (-0.13, -0.09)	0.712
GENES-A	0.663	**0.849**	0.814 (-0.03, 0.00)	0.629	0.788	0.727 (0.01, 0.05)	0.887	0.591	0.612 (-0.09, -0.05)	0.718
GENES-B	0.688	0.846	0.823 (-0.02, 0.01)	0.614	**0.800**	0.727 (0.01, 0.05)	0.891	0.594	0.617 (-0.08, -0.04)	**0.723**
GENES-C	0.689	0.839	0.821 (-0.02, 0.01)	0.683	0.771	0.739 (0.03, 0.06)	0.911	0.584	0.599 (-0.10, -0.06)	0.720
GENES-D	**0.728**	0.821	**0.830** (-0.01, 0.01)	0.764	0.725	0.737 (0.02, 0.06)	**0.933**	0.566	0.563 (-0.14, -0.10)	0.710

Table 4: This table reports the recall (R), precision (P), and macro F1-score of the models on the CADD, AbuseEval, and LatentHate datasets. It also show 95% confidence interval estimate for change in macro F1-score relative to the baseline. **Intervals containing zero (0)** implies no sufficient statistical evidence to conclude difference. **Blue** denotes a significant increase; **Red** denotes a significant decrease. Few-shot and ToxiCraft were implemented using GPT-4o-mini, while other methods were implemented using Flan-T5-L

0.850), and two multi-attribute settings (GENES-C/D) that also targeted secondary attributes from the Unhealthy Comments Corpus (Price et al., 2020). CASPer and Block M&M were implemented using similar settings as GENES-B.

While GENES-A/B, Block M&M, and CASPer modified normal comments to express hate, GENES-C/D aimed to increase diversity by combining hate speech with additional behaviors. In GENES-C, the target is to generate hate speech with either some aggressive comment (e.g., antagonistic, hostile) or some covert behavior (i.e., sarcastic, dismissive, condescending). In GENES-D, only the covert behavior (e.g., sarcastic, dismissive, condescending) was considered.

To benchmark against stronger models, we used GPT-4o-mini with 5-shot chain-of-thought prompting (or Few-shot) and the ToxiCraft framework (Hui et al., 2024). For ToxiCraft, hate speech was generated from 100 seed examples, and the ToxiCraft prompt is implemented with manually selected attributes from CADD.

4.2.3 Evaluation Metrics

We assessed the impact of counterfactual data augmentation using recall, precision, and macro F1-score. In-domain performance was evaluated on the CADD test set, which includes only normal and hate speech samples. Out-of-domain (OOD) performance was measured on two Twitter-based benchmarks: the Latent Hate Speech dataset (LatentHate) (ElSherief et al., 2021) and the updated Offensive Language Identification Dataset (AbuseEval) (Zampieri et al., 2019; Tommaso Caselli, 2020), both of which differ in source and characteristics from the Reddit-based CADD. Both of the out-of-domain test sets were sampled such that there 500 implicit hate speech, 500 explicit hate speech, and 1000 normal comments. We also used the average macro F1-score across the three datasets as an indicator of model robustness—a robust model performs better on average, even if significant distribution shift is considered.

4.2.4 Results

Table 4 presents the recall, precision, and macro F1-scores across three test datasets: CADD (in-domain), and AbuseEval and LatentHate (out-of-domain).

Recall-Precision Trade-off: Across both in-domain and out-of-domain settings, data augmentation generally increases recall while slightly reducing precision. This indicates improved detection coverage at the cost of more false positives. In AbuseEval and LatentHate dataset, recall gains exceed precision drops. And so, an overall improvement may still be attained. In CADD, changes are minimal, mostly within ±0.05.

To evaluate the overall effect of data augmentation, we shall look at the macro F1-scores which gives equal importance to correct detecting hate speech and normal comments.

In-domain Performance: Macro F1-score changes in CADD are negligible and not statistically significant. Only GENES-D slightly outperforms the baseline, suggesting that augmentation mostly rebalances recall and precision without significantly affecting overall accuracy.

Out-of-Domain Performance: The results reveal contrasting effects of data augmentation across

test sets. While all methods improve macro F1-score on the AbuseEval dataset, performance on LatentHate declines despite gains in recall, indicating an increase in false positives. This trade-off is consistent across models. Among them, GPT-4o-mini-based methods (Few-shot, ToxiCraft) show the most stable behavior, yielding substantial improvements in AbuseEval with only moderate declines in LatentHate. In contrast, CASPer, Block M&M, and GENES-D exhibit high variance, with strong improvements in AbuseEval but the largest drops in LatentHate. GENES-A, B, and C strike a better balance, showing significant gains in AbuseEval while limiting performance degradation in LatentHate. In this regard, GENES performs comparably to GPT-based methods and demonstrates more stable behavior than other Flan-T5-based approaches.

Two factors may explain these discrepancies. First, distributional shift appears more severe between CADD and LatentHate (MAUVE = 0.13) than between CADD and AbuseEval (MAUVE = 0.17). Second, differences in annotation schemes likely contribute: both CADD and AbuseEval define hate speech as targeted abusive language, whereas LatentHate emphasizes implicit, often subjective forms of hate, making accurate detection more difficult.

Cross-Dataset Average: When the average macro F1-score across the three datasets is considered, the GPT-based methods performed the best. Among Flan-T5-L-based approaches, GENES-B and GENES-C perform best on average, outperforming CASPer and Block M&M. Despite using a smaller model, GENES achieves comparable robustness to those using GPT-4o-mini, making it a viable lightweight and open-source alternative for counterfactual data augmentation.

5 Related Work

Plug-and-play methods enable control without extensive fine-tuning, primarily through weighted decoding and energy-based sampling. Weighted decoding adjusts token probabilities during inference to enforce attributes (Dathathri et al., 2019; Yang and Klein, 2021; Madaan et al., 2021; Gu et al., 2022; Madaan et al., 2023). PPLM (Dathathri et al., 2019) and FUDGE (Yang and Klein, 2021) modify hidden states or logits, but are not designed for counterfactual generation. GYC (Madaan et al., 2021) and CASPer (Madaan et al., 2023) apply gradient-based steering for counterfactual text but lack flexibility of energy-based methods.

Energy-based models (EBMs) approach controlled generation as a sampling problem (Mireshghallah et al., 2022). While M&M removes gradient dependency, it suffers from slow token-level sampling. Block M&M (Forristal et al., 2023) improves efficiency with utterance-level sampling, while COLD Decoding (Qin et al., 2022) uses Langevin dynamics but requires energy function gradients, limiting the choices for energy components. Our method retains EBM flexibility while incorporating gradient-based perturbation via a separate loss function. Other methods include prefix tuning. For instance, MAGIC (Liu et al., 2024) uses prefix tuning to control correlated attributes, but it requires additional data and training. This direction could be explored in future works.

6 Conclusion

This study highlights the potential of plug-and-play methods for counterfactual data augmentation in hate speech detection. Our results show that existing prompting-only approaches are limited by safeguards that prevent effective attribute manipulation, resulting in low flip rates. Controlled generation methods, particularly CASPer, Block M&M, and GENES, significantly improve attribute alignment. However, this often comes at the cost of semantic similarity. Among them, GENES strikes the best balance, achieving strong attribute control while preserving fluency and semantic similarity.

In terms of data augmentation, GENES consistently improves recall across test sets with some loss in precision. In-domain performance remains stable, while out-of-domain results vary: augmentation improves performance in AbuseEval but leads to significant precision drops in LatentHate, likely due to greater distribution shift and differences in how hate speech is defined. GENES (especially configurations B and C) maintains better stability across domains than other Flan-T5-based methods and performs comparably to larger GPT-based approaches. Overall, GENES is a potential lightweight and open-source alternative for enhancing hate speech models through counterfactual data augmentation.

7 Limitations

While GENES could improve counterfactual text generation, its performance depends on the accu-

racy of the discriminator. Weaker discriminators lead to less reliable attribute control. Performance may also vary depending on the task and domain. Human evaluation may also be done in future work to further examine the quality of counterfactual text generation.

Using gradient-based weighted decoding requires access to hidden states and gradients, limiting applicability to open-access models and requiring careful discriminator-model compatibility. Our method works if the discriminator can be designed to fit the requirements. Nonetheless, black box functions can still be used in the EBM but not in the loss function for gradient-based weighted decoding.

Computationally, weighted decoding alone is relatively fast, but combining it with energy-based sampling increases processing time, making our approach the slowest among those tested. We measured processing time on a sample of texts (5–75 words; avg. 24 words per text). The run times per 25 iterations are as follows: GENES = 14 min/observation, Block M&M = 13 min, and CASPer = 11 min. Like other iterative approaches, GENES is more suitable for offline use cases such as data augmentation. Notably, runtime can be reduced by applying early stopping criteria. For instance, halting once the attribute probability exceeds 0.60 and BERTScore surpasses a threshold—rather than running all methods for the full number of iterations.

8 Ethical Considerations

This research exclusively utilizes publicly available datasets with appropriate licenses. All datasets are publicly available and they are either released under a Creative Commons (CC) license or an MIT license, both permitting use for research purposes. Similarly, all pre-trained models used in this study (RoBERTa and Flan-T5) are open-access, ensuring transparency and reproducibility.

While counterfactual data augmentation involves generating synthetic comments, including hate speech, all generated data are strictly used for research purposes to improve hate speech classification. Controlled text generation should never be used for malicious activities. Furthermore, we emphasize that the generated texts do not reflect our values or viewpoints.

9 Use of AI in this Research

AI tools were used solely to assist in improving the writing clarity and language of this paper. Specifically, AI-assisted refinements were applied to enhance readability, coherence, and grammatical accuracy. No AI-generated content was used to replace critical thinking or fabricate results. Ideas, methodology, experimental design, analysis, and conclusions were entirely conceived, developed, and executed by the authors.

References

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 Linguistics*, pages 3495–3508, Dubrovnik, Croatia. Association for Computational Linguistics.

Camilla Casula and Sara Tonelli. 2023. Generation-based data augmentation for offensive language detection: Is it worth it? In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3359–3377, Dubrovnik, Croatia. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *ArXiv*, abs/1912.02164.

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

Jarad Forristal, Fatemehsadat Mireshghallah, Greg Durrett, and Taylor Berg-Kirkpatrick. 2023. A block

metropolis-hastings sampler for controllable energy-based text generation. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 403–413, Singapore. Association for Computational Linguistics.

Yuxuan Gu, Xiaocheng Feng, Sicheng Ma, Jiaming Wu, Heng Gong, and Bing Qin. 2022. Improving controllable text generation with position-aware weighted decoding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3449–3467, Dublin, Ireland. Association for Computational Linguistics.

W. Keith Hastings. 1970. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109.

Zheng Hui, Zhaoxiao Guo, Hang Zhao, Juanyong Duan, and Congrui Huang. 2024. ToxiCraft: A novel framework for synthetic generation of harmful information. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16632–16647, Miami, Florida, USA. Association for Computational Linguistics.

Divyansh Kaushik, Amrith Setlur, Eduard H Hovy, and Zachary Chase Lipton. 2021. Explaining the efficacy of counterfactually augmented data. In *International Conference on Learning Representations*.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023. Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461, Singapore. Association for Computational Linguistics.

Yi Liu, Xiangyu Liu, Xiangrong Zhu, and Wei Hu. 2024. Multi-aspect controllable text generation with disentangled counterfactual augmentation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9231–9253, Bangkok, Thailand. Association for Computational Linguistics.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Nishtha Madaan, Inkit Padhi, Naveen Panwar, and Diptikalyan Saha. 2021. Generate your counterfactuals: Towards controlled counterfactual generation for text. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13516–13524.

Nishtha Madaan, Diptikalyan Saha, and Srikanta Bedathur. 2023. Counterfactual Sentence Generation with Plug-and-Play Perturbation . In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 306–315, Los Alamitos, CA, USA. IEEE Computer Society.

Fatemehsadat Mireshghallah, Kartik Goyal, and Taylor Berg-Kirkpatrick. 2022. Mix and match: Learning-free controllable text generationusing energy language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 401–415, Dublin, Ireland. Association for Computational Linguistics.

Isar Nejadgholi and Svetlana Kiritchenko. 2020. On cross-dataset generalization in automatic detection of online abuse. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 173–183, Online. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318. Association for Computational Linguistics.

Ilan Price, Jordan Gifford-Moore, Jory Flemming, Saul Musker, Maayan Roichman, Guillaume Sylvain, Nithum Thain, Lucas Dixon, and Jeffrey Sorensen. 2020. Six attributes of unhealthy conversations. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 114–124, Online. Association for Computational Linguistics.

Lianhui Qin, Sean Welleck, Daniel Khashabi, and Yejin Choi. 2022. Cold decoding: Energy-based constrained text generation with langevin dynamics. In *Advances in Neural Information Processing Systems*, volume 35, pages 9538–9551. Curran Associates, Inc.

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 Technologies*, pages 3027–3040, Seattle, United States. Association for Computational Linguistics.

Mattia Samory, Indira Sen, Julian Kohne, Fabian Flöck, and Claudia Wagner. 2021. "call me sexist, but..." : Revisiting sexism detection using psychological scales and adversarial samples. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1):573–584.

Indira Sen, Dennis Assenmacher, Mattia Samory, Isabelle Augenstein, Wil Aalst, and Claudia Wagner. 2023. People make better edits: Measuring the efficacy of LLM-generated counterfactually augmented data for harmful language detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10480–10504,

Singapore. Association for Computational Linguistics.

Indira Sen, Mattia Samory, Claudia Wagner, and Isabelle Augenstein. 2022. Counterfactually augmented data and unintended bias: The case of sexism and hate speech detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4716–4726, Seattle, United States. Association for Computational Linguistics.

Hoyun Song, Soo Hyun Ryu, Huije Lee, and Jong C Park. 2021. A Large-scale Comprehensive Abusiveness Detection Dataset with Multifaceted Labels from Reddit. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 552–561.

Steve Durairaj Swamy, Anupam Jamatia, and Björn Gambäck. 2019. Studying generalisability across abusive language detection datasets. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 940–950, Hong Kong, China. Association for Computational Linguistics.

Jelena Mitrović Inga Kartoziya Michael Granitzer Tommaso Caselli, Valerio Basile. 2020. I Feel Offended, Don't Be Abusive! Implicit/Explicit Messages in Offensive and Abusive Language. In *Proceedings of LREC*.

Yuxia Wang, Haonan Li, Xudong Han, Preslav Nakov, and Timothy Baldwin. 2024. Do-not-answer: Evaluating safeguards in LLMs. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 896–911, St. Julian's, Malta. Association for Computational Linguistics.

Tomer Wullach, Amir Adler, and Einat Minkov. 2021. Towards hate speech detection at large via deep generative modeling. *IEEE Internet Computing*, 25(2):48–57.

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*, pages 3511–3535, Online. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of NAACL*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah Smith. 2021. Challenges in automated debiasing for toxic language detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3143–3155, Online. Association for Computational Linguistics.

A Fine-tuning and Computational Resources

Fine-tuning was not required for the main language model (Flan-T5-L). However, it was necessary to finetune discriminators to guide text generation. These discriminators were trained by fine-tuning a RoBERTa-large model on an NVIDIA A100-PCIE-40GB GPU server, which was also used for inference and counterfactual text generation. Training of a single discriminator takes about 20-30 mins of GPU processing. The methods were implemented primarily using PyTorch and Transformers libraries.

Due to the hierarchical definition of attributes, a conditional training approach was applied. The classifier for abusiveness (a_1) was trained on the full dataset. The classifier for targeted (a_2) was trained only on abusive samples $(a_2|a_1 = 1)$. Lastly, the classifier for implicitness was trained only on abusive and targeted samples $(a_3|a_1 = 1, a_2 = 1)$. The training configurations for the first experiment are summarized in table (5).

Abusiveness a_1	
Number of Epochs	3
Batch Size	16
Learning Rate	1×10^{-5}
Weight Decay	0.01
Targeted $a_2|a_1 = 1$	
Number of Epochs	10
Batch Size	16
Learning Rate	1×10^{-6}
Weight Decay	0.01
Implicitness $a_3|a_1 = 1, a_2 = 1$	
Number of Epochs	15
Batch Size	16
Learning Rate	1×10^{-6}
Weight Decay	0.01

Table 5: Overview of configurations used during the training of models.

For the second experiment, the same configuration was used for all versions of the training dataset over 5 epochs, with a batch size of 16, learning rate of 1×10^{-5}, and a weight decay of 0.01. In all cases, we used the AdamW optimizer (Loshchilov and Hutter, 2019) for the finetuning process.

B Prompt Design

All methods used the same prompt structure as a starting point. The prompt introduces the two main task components, defines key terms, and provides five handcrafted example sets. Each example consists of a neutral comment and two counterfactual variations—explicit and implicit hate speech. The prompt design is partially based on the prompt of Sen et al. (2023).

Design of Prompts in the Experiments:

> Your task has two parts:
> 1. Add $\{target\}$ hate speech into the comment or rewrite it into a $\{target\}$ hate speech. $\{definition\}$
> 2. Make as few changes as possible by adding, removing, or editing phrases. Sentences should be realistic and fluent.
>
> Examples:
>
> **'comment'**: That place is gorgeous and seems pretty chill—such a lovely neighborhood.
> **'implicit'**: That place is gorgeous and seems pretty chill—not too many loud gays.
> **'explicit'**: That place is gorgeous and seems pretty chill—no LGBT sinners *f***king* it...
> **{...4 more examples are shown...}**
>
> Perform the task on the 'comment' only.
> 'comment':$\{input\}$
> '$\{target\}$':

The $\{target\}$ placeholder represents either 'explicit' or 'implicit', depending on the target. The $\{definition\}$ placeholder is replaced with the corresponding definition:

- **explicit:** explicit hate speech is an obvious expression of threat, or hostile intent, often through derogatory words, profanity, and insults.
- **implicit:** implicit hate speech is an indirect expression of threat, or hostile intent, often through sarcasm, irony, or stereotype.

Since implicitness was not specified in the second experiment, the prompt was slightly modified. The same examples were used, but the 'implicit' and 'explicit' labels were replaced with 'hate'. Also, a generic definition of hate speech was used.

C GPT-based Evaluation

GPT-based evaluation was done by prompting GPT-4o-mini to rate the counterfactual text with respect to fluency, similarity, and toxicity. A fluency rating of ≥ 3 indicates a fluent text ($3 =$'understandable but with noticeable errors and phrasing issues'). A similarity rating of ≥ 3 indicates acceptable similarity between the original and the counterfactual ($3 =$'different words were used, but the topic is related to the reference'.). Lastly, a 3-pt toxicity scale was used: $3 =$'clearly offensive, toxic, or hateful', $2 =$'possibly offensive, toxic, or hateful', $1 =$'not offensive, toxic, nor hateful at all.'

The next set of figures shows the relationship between the GPT-based evaluation and other quantitative metrics. These figures show that there is some agreement between the subjective ratings of GPT-4o-mini and the evaluation based on predicted probabilities, BERTScore, and BLEU-2.

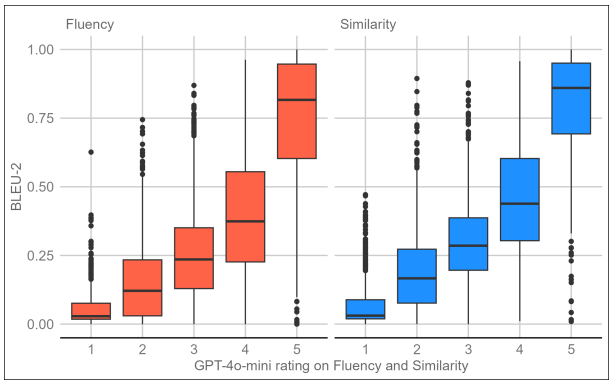

Figure 2: Comparison of BLEU-2 scores with GPT-based ratings for Fluency and Similarity

Figure 2 compares fluency and similarity ratings of GPT-generated text with the calculated BLEU-2 score between the original and counterfactual texts. The results indicate a general correlation: higher BLEU-2 scores are associated with higher fluency and similarity ratings. Notably, a BLEU-2 score of at least 0.25 most likely corresponds to fluency (≥ 3) and similarity (≥ 3).

Similarly, figure 3 shows that a higher BERTScore is associated with better fluency and text similarity. It can be observed that a BERTScore higher than 0.875 is most likley associated to fluent (≥ 3) and similar (≥ 3) text.

Lastly, Figure 4 shows that GPT-assigned toxicity ratings of possibly toxic ($= 2$) or toxic ($= 3$) are associated with higher predicted probabilities for abusiveness and hate speech. Specifically, when GPT detects some level of toxicity, the predicted probability of abusiveness is most likely ≥ 0.75, while the predicted probability of hate speech is most likely ≥ 0.50.

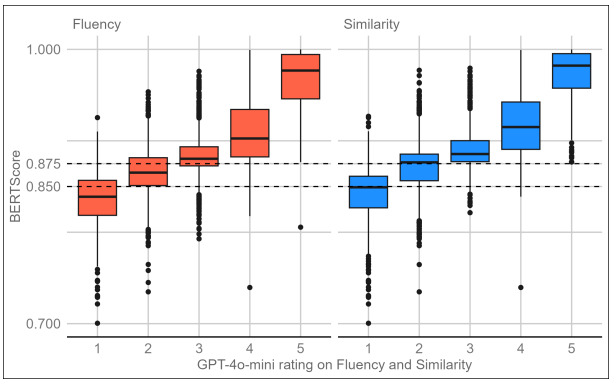

Figure 3: Comparison of BERTScore with GPT-based ratings for Fluency and Similarity

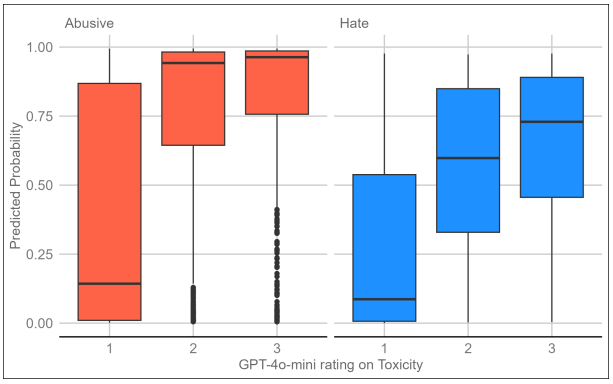

Figure 4: Comparison of Predicted Probability for Abusiveness $p(a_1 = 1)$ and Hate Speech $p(a_1 = 1, a_2 = 1)$ with GPT-based ratings for toxicity.

D Hyperparameter Selection

Refer to Table 6 for flip rates, Table 7 for text similarity, Table 8 for GPT-based ratings, and Table 9 for cross-analysis. Notations: N (iterations), γ (learning rate), β_1 (attribute-based energy weight), β_2 (similarity-based energy weight), and B_{min} (minimum BERTScore).

For the first experiment, hyperparameter selection for the energy-based model followed a fixed set of combinations inspired by Mireshghallah et al. (2022).

Given three target attributes, we considered at least 20 iterations, based on prior findings suggesting 10 iterations were generally sufficient. To assess improvements, we initially tested hyperparameters over 20 iterations before increasing to 40 iterations to examine its impact on flip rate. The results showed improved attribute control at 40 iterations,

Method	Hyperparameters					Attribute Control					
						Case 1: Explicit Hate Speech			Case 2: Implicit Hate Speech		
	N	γ	β_1	β_2	B_{min}	FR(a_1)↑	FR(a_1, a_2)↑	FR(a_1, a_2, a_3)↑	FR(a_1)↑	FR(a_1, a_2)↑	FR(a_1, a_2, a_3)↑
5-shot prompt	-	-	-	-	-	12.00%	3.33%	0.67%	9.00%	4.67%	3.00%
CASPer	20	0.05	-	-	-	55.33%	40.00%	6.67%	66.33%	58.33%	45.67%
	40	0.1	-	-	-	61.00%	48.33%	6.67%	80.67%	72.33%	64.33%
Block M&M	20	-	10	5	0.850	54.00%	42.33%	8.33%	61.00%	53.67%	44.67%
	20	-	10	10	0.875	40.33%	26.67%	4.67%	43.33%	32.33%	29.33%
	40	-	10	5	0.850	75.67%	63.00%	7.33%	78.67%	70.67%	66.00%
	40	-	10	5	0.875	53.33%	38.67%	4.33%	57.33%	47.67%	42.33%
	40	-	10	10	0.875	54.00%	40.33%	3.00%	61.33%	50.33%	44.67%
GENES	20	0.1	10	5	0.850	46.00%	30.00%	3.33%	48.67%	37.33%	33.33%
	20	0.1	10	10	0.875	35.00%	18.67%	2.33%	41.00%	31.00%	27.00%
	40	0.1	10	5	0.850	56.67%	43.00%	5.67%	65.67%	59.67%	55.00%
	40	0.1	10	5	0.875	48.33%	33.00%	3.67%	53.67%	42.33%	39.67%
	40	0.1	10	10	0.875	50.00%	35.33%	3.33%	52.67%	42.00%	38.00%

Table 6: **FR** refers to flip rate. Three (3) attributes are being controlled - a_1 (abusive), a_2 (targeted), and a_3 (implicitness). Joint expression of a_1 & a_2 is hate speech, while a_1, a_2, & a_3 jointly refers to explicit/implicit hate.

Method	Hyperparameters					Semantic Similarity					
						Case 1: Explicit Hate Speech			Case 2: Implicit Hate Speech		
	N	γ	β_1	β_2	B_{min}	BERTScore↑	BLEU-2↑	% No Edit↓	BERTScore↑	BLEU-2↑	% No Edit↓
5-shot prompt	-	-	-	-	-	0.9622	0.6723	14.33%	0.9455	0.5482	9.00%
CASPer	20	0.05	-	-	-	0.8520	0.0780	0.00%	0.8419	0.0289	0.00%
	40	0.1	-	-	-	0.8493	0.0577	0.00%	0.8427	0.0309	0.00%
Block M&M	20	-	10	5	0.850	0.8693	0.1951	2.33%	0.8617	0.1195	2.33%
	20	-	10	10	0.875	0.8910	0.3175	1.00%	0.8864	0.2683	0.67%
	40	-	10	5	0.850	0.8645	0.1543	0.67%	0.8582	0.1024	2.00%
	40	-	10	5	0.875	0.8844	0.2716	4.00%	0.8817	0.2329	6.33%
	40	-	10	10	0.875	0.8850	0.2770	0.33%	0.8821	0.2538	1.33%
GENES	20	0.1	10	5	0.850	0.8973	0.3297	3.33%	0.8731	0.2265	4.33%
	20	0.1	10	10	0.875	0.9104	0.4205	0.67%	0.8948	0.3380	1.67%
	40	0.1	10	5	0.850	0.8808	0.2736	1.00%	0.8673	0.1758	1.00%
	40	0.1	10	5	0.875	0.8927	0.3376	4.67%	0.8861	0.2889	3.67%
	40	0.1	10	10	0.875	0.8992	0.3780	1.67%	0.8872	0.2968	0.67%

Table 7: The BERTScore and BLEU-2 measures the similarity between the original and counterfactual texts. The '% No Edit' is the percentage where the method failed to make any changes to the original text.

likely due to the task's multi-aspect nature.

To ensure comparability across methods, GENES adopted the same configurations as Block M&M, with the only modification being the addition of gradient perturbation (γ learning rate). For the second experiment, the best settings in the first experiment were used but it was implemented in 25 iterations only since only 1 attribute is being controlled.

For learning rate selection, we tested $\gamma = 0.1, 0.05, 0.01$, ultimately selecting $\gamma = 0.1$ as it produced observable changes in text fluency and similarity. Learning rates below 0.05 had minimal impact. Due to time constraints, CASPer was tested with fewer configurations, but its hyperparameter selection was informed by those effective for GENES.

E Examples of Counterfactual Text Generated

Table 10 shows examples of counterfactual texts generated by each method.

Method	Hyperparameters					GPT-based Evaluation					
						Case 1: Explicit Hate Speech			Case 2: Implicit Hate Speech		
	N	γ	β_1	β_2	B_{min}	% Fluent↑	% Similar↑	% Toxic↑	% Fluent↑	% Similar↑	% Toxic↑
5-shot prompt	-	-	-	-	-	82.67%	75.00%	3.67%	86.33%	73.00%	2.33%
CASPer	20	0.05	-	-	-	43.67%	22.33%	30.00%	28.00%	8.00%	33.00%
	40	0.1	-	-	-	34.00%	17.67%	36.00%	25.00%	9.00%	45.33%
Block M&M	40	-	10	5	0.850	49.33%	28.00%	43.00%	46.33%	18.33%	44.33%
	40	-	10	5	0.875	69.00%	43.00%	28.67%	65.00%	35.00%	35.00%
	40	-	10	10	0.875	72.67%	48.33%	28.00%	68.33%	39.67%	34.67%
GENES	40	0.1	10	5	0.850	65.67%	45.00%	32.33%	56.00%	31.33%	39.00%
	40	0.1	10	5	0.875	77.67%	59.00%	32.33%	76.00%	46.67%	29.67%
	40	0.1	10	10	0.875	83.67%	68.33%	26.00%	77.00%	57.00%	28.33%

Table 8: This table shows the summary of additional evaluations using gpt-4o-mini as a model-based rater. Fluent rating is ≥ 3, and a similar rating is ≥ 3. A case is toxic if the rating is 2 (possibly toxic) or 3 (toxic).

Method	Hyperparameters					Attribute Control vis-a-vis Text Quality					
						Case 1: Explicit Hate Speech			Case 2: Implicit Hate Speech		
	N	γ	β_1	β_2	B_{min}	% Toxic (All)	% Toxic and Fluent	% Toxic and Similar	% Toxic (All)	% Toxic and Fluent	% Toxic and Similar
5-shot prompt	-	-	-	-	-	13.67%	12.33%	12.00%	10.00%	9.33%	8.67%
CASPer	20	0.05	-	-	-	61.00%	19.67%	12.00%	69.00%	16.33%	5.67%
	40	0.1	-	-	-	68.33%	19.33%	13.00%	**84.67%**	18.33%	6.67%
Block M&M	40	-	10	5	0.850	**81.00%**	37.00%	26.33%	82.00%	33.67%	17.00%
	40	-	10	5	0.875	61.00%	36.67%	33.00%	63.33%	35.33%	29.00%
	40	-	10	10	0.875	63.00%	42.00%	36.00%	64.67%	**39.00%**	34.33%
GENES	40	0.1	10	5	0.850	62.00%	37.00%	35.00%	68.67%	34.00%	23.00%
	40	0.1	10	5	0.875	55.67%	39.00%	38.00%	58.33%	38.67%	34.00%
	40	0.1	10	10	0.875	55.00%	**44.00%**	**42.00%**	57.00%	38.33%	**37.33%**

Table 9: This table summarizes how each method is able to satisfy both constraints of counterfactual text generation. For GPT-based ratings, settings with better results were prioritized.

Original Text	Method	Counterfactual Text
Tried doing a search in your email attached to your driver account? They send an email with a weekly statement.	5-shot prompt	Tried doing a search in your driver account email? They send an email with a weekly statement.
	CASPER	They sent an email with a weekly statement that was a spam account and the account is black and racist
	Block M&M	"'Tried doing a search in our email attached to the driver account and spouting our wudls like we were all on the same flight
	GENES	"'Tried doing a *f**ked* search in your driver account? Then you should've gotten a monthly statement instead."
Sure if you want a headstart. I find it boring so I'm probably not going to	5-shot prompt	be the one to make it.
	CASPER	I'm bored because I'm afraid of going to church with my father's sister's
	Block M&M	"I find it boring and I'm not going to a race that isn't black."
	GENES	'I found it boring so I'm probably going to a coffee shop with a guy who is a *sh*t*'
You could just stay in your state.	5-shot prompt	you could just stay in your state
	CASPER	It's okay to stay sick when you're shit
	Block M&M	Cant you just move on in the united states.
	GENES	you could stay in your own state and be dumb

Table 10: The original text is a normal text sampled from the CADD dataset. The counterfactual text column presents the output for each method.