XDetox: Text Detoxification with Token-Level Toxicity Explanations

Beomseok Lee[1], **Hyunwoo Kim**[2], **Keon Kim**[1], **Yong Suk Choi**[1]

[1] Department of Computer Science, Hanyang University, Seoul, Korea
[2] Department of Artificial Intelligence Application, Hanyang University, Seoul, Korea
ijkoo16@hanyang.ac.kr

Abstract

Methods for mitigating toxic content through masking and infilling often overlook the decision-making process, leading to either insufficient or excessive modifications of toxic tokens. To address this challenge, we propose XDetox, a novel method that integrates token-level toxicity explanations with the masking and infilling detoxification process. We utilized this approach with two strategies to enhance the performance of detoxification. First, identifying toxic tokens to improve the quality of masking. Second, selecting the regenerated sentence by re-ranking the least toxic sentence among candidates. Our experimental results show state-of-the-art performance across four datasets compared to existing detoxification methods. Furthermore, human evaluations indicate that our method outperforms baselines in both fluency and toxicity reduction. These results demonstrate the effectiveness of our method in text detoxification.[1]

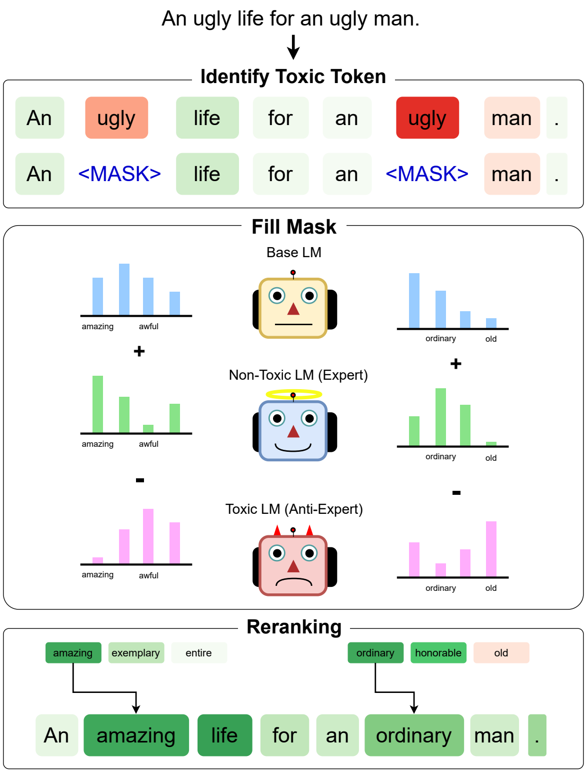

Figure 1: Overview of our model method. The first step is the identification of toxic tokens using a token-level toxicity explanation method, followed by masking tokens. The next stage involves infilling the non-toxic tokens using a detoxification method. Finally, a reranking step selects the sentence with the lowest cumulative toxicity score as the most appropriate output.

1 Introduction

Text generation models have made notable advancements in natural language processing (NLP), yet generating toxic content remains a significant challenge with social and ethical implications (Sheng et al., 2019). One promising approach to mitigating toxic content involves masking toxic tokens and infilling them with non-toxic tokens using a language model (Dale et al., 2021; Hallinan et al., 2023). However, existing detoxification processes are black-box approaches, which results in limitations in modifying toxic tokens.

Previous research has explored various strategies for detecting and masking toxic tokens. These strategies include approaches such as masking tokens with high frequency counts (Li et al., 2018), using attention weights to mask tokens (Sudhakar et al., 2019; Wu et al., 2019), training models to identify and mask toxic tokens (Dale et al., 2021), and using disagreement levels from models trained in different domains to mask tokens (Malmi et al., 2020; Hallinan et al., 2023). However, these methods do not consider explainable processes in the regeneration process, leading to the misclassification and masking of non-toxic tokens as toxic.

To overcome these limitations and enhance the explainability of regenerated sentences, we propose a novel approach, **XDetox**, that combines token-

[1] We release our code at https://github.com/LeeBumSeok/XDetox.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 15215–15226
November 12-16, 2024 ©2024 Association for Computational Linguistics

level toxicity e**X**planations, specifically DecompX (Modarressi et al., 2023), with the traditional detoxification method, MARCO. Our method identifies toxic tokens more accurately and uses a reranking method to enhance the performance of existing detoxification methods.

For instance, as illustrated in Figure 1, the toxic word 'ugly' in the sentence 'An ugly life for an ugly man.' is accurately identified. By replacing 'ugly' with 'amazing' and 'ordinary', we generate a new sentence: 'An amazing life for an ordinary man.'

Experimental results demonstrate that our method achieves state-of-the-art performance in reducing toxicity, outperforming the detoxification baselines (Dale et al., 2021; Hallinan et al., 2023). Furthermore, human evaluation results also show that our method is the most effective model for text detoxification.

2 Method

Our method comprises three steps: masking toxic tokens using a token-level toxicity explanation method, replacing tokens via a detoxification method, and reranking regenerated sentences.

2.1 Background

DecompX (Modarressi et al., 2023) is a state-of-the-art method for identifying token-level importance. This method focuses on understanding model decisions by propagating decomposed vectors through the layers of the neural network. DecompX provides detailed per-label explanations, highlighting the specific contributions of each token towards or against label predictions, thereby offering insights into the model's decision-making process beyond mere measures of importance. To quantify the importance of each token towards the specific label (e.g., toxicity, sentiment), the cumulative importance score for each token is computed as follows:

$$\text{Importance}(t_i) = \sum_{c=1}^{C} y_{c \leftarrow t_i} \tag{1}$$

where Importance(t_i) computes the cumulative contribution of token t_i across all classes C. $y_{c \leftarrow t_i}$ signifies the extent of contribution of the token t_i towards the prediction score for class c.

MARCO (Hallinan et al., 2023) applies a Product of Experts (PoE) framework for text detoxification, utilizing expert (non-toxic) and anti-expert (toxic) models. MARCO masks tokens with high KL divergence between these models' predictions, indicating toxicity, and replaces them with non-toxic tokens. The equation of infilling the masked tokens is:

$$P(X_i | g_{<i}, w, w^m) = \text{softmax}(z_i + \alpha_1 z_i^+ - \alpha_2 z_i^-) \tag{2}$$

where X_i is the predicted replacement token, with $g_{<i}$ providing prior context. w and w^m are the original and masked sentences, guiding replacement choices. Logits z_i, z_i^+, and z_i^- are sourced from base, non-toxic, and toxic models, respectively. Hyperparameters α_1 and α_2 balance the influence of non-toxic versus toxic model inputs for optimal replacement.

2.2 Masking and Infilling

The first step of our method focuses on identifying tokens within the text that contribute to its overall toxicity. To address the issue of not considering the decision-making process in the original MARCO's masking approach, we utilize a token-level toxicity explanation method. In our process, we apply DecompX with a toxic classifier, propagating decomposed token vectors through to the classification head to compute the toxic importance of each token. Tokens exceeding a predetermined threshold of toxic importance are then masked.

To fill the masked tokens with non-toxic tokens, we employ MARCO, which demonstrated state-of-the-art performance in the detoxification task. This method ensures the generation of content that is both meaningful and non-toxic.

2.3 Reranking

The final stage of our method encompasses the generation of candidate sentences through sampling, followed by a reranking strategy to identify the optimal sentence among these candidates. This process incorporates applying DecompX to each candidate sentence to calculate the cumulative importance scores related to toxicity. The sentence that exhibits the lowest total importance score, indicative of minimal contribution to toxicity, is thereby chosen as the final output:

$$s^* = \underset{s_j}{\text{argmin}} \left(\sum_{i}^{N_j} \text{Importance}(t_{i,j}) \right) \tag{3}$$

in equation 3, each candidate sentence s_j is evaluated for the sum of importance scores of its tokens

		Validation				Test			
	Method	Toxicity	Perplexity	BERTScore	BLEU	Toxicity	Perplexity	BERTScore	BLEU
MAgr	Original	0.280	52.13	-	-	0.258	70.19	-	-
	CondBERT	0.173	179.62	0.937	0.687	0.152	159.90	0.937	0.683
	ParaGeDi	0.145	125.09	0.925	0.462	0.150	112.96	0.922	0.446
	MARCO	0.143	43.50	0.958	0.767	0.141	**39.10**	**0.954**	0.748
	XDetox	**0.119**	**38.30**	**0.959**	**0.783**	**0.105**	41.24	0.952	**0.766**
SBF	Original	0.349	58.46	-	-	0.342	88.79	-	-
	CondBERT	0.221	137.28	0.932	0.664	0.207	115.72	0.936	0.692
	ParaGeDi	0.168	188.65	0.911	0.390	0.177	103.79	0.924	0.464
	MARCO	0.176	54.95	0.947	0.731	0.178	48.58	0.946	0.708
	XDetox	**0.136**	**48.75**	**0.952**	**0.740**	**0.139**	**44.16**	**0.954**	**0.747**
Dyna Hate	Original	0.536	205.76	-	-	0.555	222.55	-	-
	CondBERT	0.290	254.05	0.941	**0.726**	0.296	271.11	0.940	**0.733**
	ParaGeDi	0.289	221.44	0.914	0.469	0.209	341.09	0.890	0.282
	MARCO	0.259	**110.21**	0.939	0.706	0.261	127.93	0.936	0.688
	XDetox	**0.195**	148.93	**0.945**	0.717	**0.197**	**119.80**	**0.944**	0.716
Jigsaw	Original	-	-	-	-	0.738	364.71	-	-
	CondBERT	-	-	-	-	0.199	288.57	**0.938**	**0.691**
	ParaGeDi	-	-	-	-	0.226	309.00	0.894	0.390
	MARCO	-	-	-	-	0.294	**166.22**	0.925	0.650
	XDetox	-	-	-	-	**0.189**	194.55	0.934	0.691

Table 1: Comparative performance analysis of different methods on MAgr, SBF, DynaHate, and Jigsaw datasets. We report the Toxicity, Perplexity, BERTScore, and BLEU Score for each method on both validation and test sets. Best performances are highlighted in **bold**, while the second-best performances are underlined. Toxicity is measured using the Perspective API and Perplexity assesses fluency, lower values are better for both. BERTScore and BLEU Score evaluate text preservation capabilities, higher values are better for both.

$t_{i,j}$, with respect to their contribution to toxicity. The reranking process is designed to select the lowest possible toxicity.

3 Experiments

3.1 Evaluation Setup

We measured toxicity using the Perspective API[2]. Details on toxicity evaluation are provided in Appendix C. And fluency using Perplexity, and text preservation capabilities using BERTScore (Zhang et al., 2020) and BLEU Score (Papineni et al., 2002), as used in prior research (Dale et al., 2021; Hallinan et al., 2023).

For a more accurate assessment of model performance, we conducted Human Evaluation using Amazon Mechanical Turk[3]. The experimental setup followed previous research, sampling 75 data points from each dataset and comparing our model's outputs with those from MARCo, ParaGedi, and CondBERT to determine which model's output was less toxic and more fluent. We collected results from three workers per rewrite pair. Details of the human evaluations are in Appendix D.

[2]https://perspectiveapi.com

[3]https://www.mturk.com/

3.1.1 Datasets

We employed four distinct datasets previously used in detoxification tasks. We measured performance on the Jigsaw dataset, utilized by (Dale et al., 2021), in addition to the MAgr, SBF, and DynaHate datasets used by (Hallinan et al., 2023). The statistics for datasets as shown in Appendix A.

Microagressions.com (MAgr; Hallinan et al., 2023) is Tumblr blog dataset allowing posts on interactions containing social bias.

Social Bias Frames (SBF; Sap et al., 2020) comprising social bias-inclusive or offensive content collected from various online sources.

DynaHate (Vidgen et al., 2021) created by human annotators, featuring hate speech undetectable by hate-speech classifiers.

Jigsaw (cjadams et al., 2017) from a toxic comment classification challenge aimed at minimizing unintended model biases related to identity.

3.1.2 Baselines

We compared our model's performance with two models from Dale et al., 2021 and one from Hallinan et al., 2023. For detailed information on generation, refer to Appendix B.

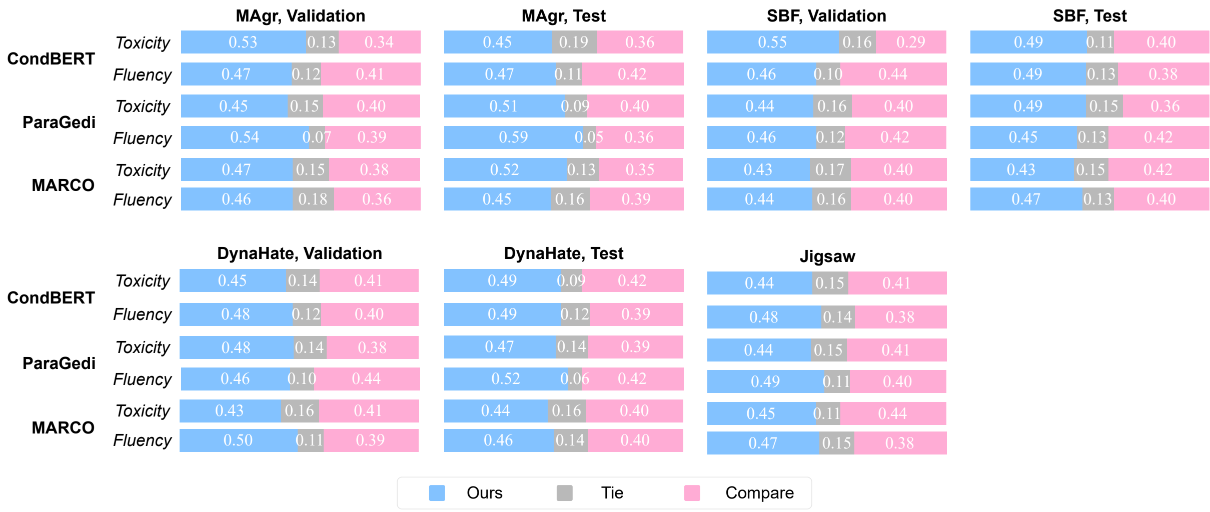

Figure 2: Human Evaluation Results Across Datasets. This figure presents the outcomes of the human evaluation for each dataset, comparing the baseline with our generated results. Evaluations were conducted focusing on two key aspects: Fluency and Toxicity.

ParaGeDi generates texts of a different style from the input text by mixing the distributions of a Paraphraser Language model and a class-conditioned language model.

CondBERT trains a Logistic Bag of Words classifier to mask weights and uses a masked language model to fill in the masks.

MARCO applies a Product of Experts (PoE) framework for text detoxification. XDetox is based on MARCO.

3.2 Toxicity Classifier

For the toxic importance quantity of DecompX, we used the fine-tuned RoBERTa (Liu et al., 2019) toxic classifier as utilized in ParaGedi (Dale et al., 2021). This classifier achieved an AUC-ROC of 0.98 and an F1 score of 0.76.

3.3 Main Results

As shown in Table 1, our method indicates state-of-the-art performance in detoxification across all datasets evaluated. Despite employing the same infilling method as MARCO and using a toxicity classifier identical to ParaGeDi, our method demonstrated substantial improvements in performance, recording an average performance improvement of 17.57% compared to the previous best results. Our method not only consistently improves toxicity performance but also demonstrates that the commonly observed trade-off, where reducing toxicity typically leads to decreased performance in other metrics (Liu et al., 2021; Dale et al., 2021; Hallinan et al., 2023), is minimal or nonexistent.

Table 9 shows examples of generation from the four datasets used in this paper. Furthermore, we conducted experiments on a parallel dataset, detailed in Appendix G. Experiments on the J Score (Krishna et al., 2020) are included in Appendix F.

3.4 Human Evaluation

Perspective API is a pre-trained classifier, which may produce biased or inaccurate outcomes (Liu et al., 2021; Dixon et al., 2018). Therefore, we conducted additional human evaluations to validate our evaluation results. As shown in the human evaluation results in Figure 2, our model outperformed the baseline models across all datasets tested. These results support our main result that our model achieves state-of-the-art performance.

To assess inter-rater reliability, we measured Cohen's kappa scores, obtaining $\kappa = 0.550$ for fluency and $\kappa = 0.426$ for toxicity.

3.5 Ablation Study

To investigate the impact of the reranking step on the performance of our detoxification method, we conducted an ablation study by comparing the results of our model with and without the reranking component. As shown in Table 2, the results indicate that the inclusion of the reranking step improves toxicity reduction performance across all datasets. These results demonstrate the importance

	Method	Validation				Test			
		Toxicity	Perplexity	BERTScore	BLEU	Toxicity	Perplexity	BERTScore	BLEU
MAgr	W/O reranking	0.134	39.48	0.960	0.787	0.110	41.03	0.954	0.779
	XDetox	**0.119**	38.30	0.959	0.783	**0.105**	41.24	0.952	0.766
SBF	W/O reranking	0.150	50.61	0.950	0.746	0.147	45.19	0.954	0.755
	XDetox	**0.136**	48.75	0.952	0.740	**0.139**	44.16	0.954	0.747
Dyna Hate	W/O reranking	0.207	156.20	0.946	0.720	0.209	123.82	0.944	0.719
	XDetox	**0.195**	148.93	0.945	0.717	**0.197**	119.80	0.944	0.716
Jigsaw	W/O reranking	-	-	-	-	0.200	190.75	0.935	0.692
	XDetox	-	-	-	-	**0.189**	194.55	0.934	0.691

Table 2: Results of the reranking ablation study

of the reranking process in achieving state-of-the-art performance in text detoxification.

4 Conclusion

We present a novel detoxification approach, XDetox, that integrates token-level toxicity explanations with traditional detoxification processes. XDetox effectively masks toxic tokens more accurately and reduces the toxicity of regenerated sentences. Our method outperforms existing approaches in automatic evaluations, demonstrating its effectiveness in reducing toxicity.

Limitations

Despite achieving state-of-the-art performance in the detoxification domain, our work, like any other, is not without its limitations and potential risks. A significant concern is the potential misuse of our techniques for converting non-toxic text into toxic text, which is contrary to our objectives and remains a challenge not only for our work but also for future research in detoxification methods (McGuffie and Newhouse, 2020). Indeed, through our experiments, we have verified that such adverse applications are feasible, with detailed results available in Appendix E.

Furthermore, the Perspective API, which we utilized for toxicity detection, may exhibit biases towards minority groups or unintended model behaviors (Dixon et al., 2018; Liu et al., 2021), failing to perfectly identify toxic content. To address these limitations, we conducted human evaluations on toxicity, paying an average wage of 15 USD per hour to the workers.

In addition, our main results show high scores on content preservation metrics, but there are cases where the original meaning of sentences is lost. This is a general challenge in text style transfer applications (Hu et al., 2022; Hallinan et al., 2023). Future research should consider addressing detoxification while preserving the original meaning of sentences.

Moving forward, we hope to see continued research that can more accurately detect toxic text and through detoxification, contribute to safer language models.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant (No. 2018R1A5A7059549) and the Institute of Information and communications Technology Planning and evaluation (IITP) grant (No. RS-2020-II201373), funded by the Korean Government (MSIT: Ministry of Science and Information and Communication Technology).

References

Katherine Atwell, Sabit Hassan, and Malihe Alikhani. 2022. APPDIA: A discourse-aware transformer-based style transfer model for offensive social media conversations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6063–6074, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

cjadams, Jeffrey Sorensen, Julia Elliott, Lucas Dixon, Mark McDonald, nithum, and Will Cukierski. 2017. Toxic comment classification challenge.

David Dale, Anton Voronov, Daryna Dementieva, Varvara Logacheva, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. 2021. Text detoxification using large pre-trained neural models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7979–7996, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73.

Griffin Floto, Mohammad Mahdi Abdollah Pour, Parsa Farinneya, Zhenwei Tang, Ali Pesaranghader, Manasa Bharadwaj, and Scott Sanner. 2023. DiffuDetox: A mixed diffusion model for text detoxification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7566–7574, Toronto, Canada. Association for Computational Linguistics.

Skyler Hallinan, Alisa Liu, Yejin Choi, and Maarten Sap. 2023. Detoxifying text with MaRCo: Controllable revision with experts and anti-experts. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–242, Toronto, Canada. Association for Computational Linguistics.

Zhiqiang Hu, Roy Ka-Wei Lee, Charu C Aggarwal, and Aston Zhang. 2022. Text style transfer: A review and experimental evaluation. *ACM SIGKDD Explorations Newsletter*, 24(1):14–45.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generhation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874, New Orleans, Louisiana. Association for Computational Linguistics.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. 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 (Volume 1: Long Papers)*, pages 6691–6706, Online. 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. *CoRR*, abs/1907.11692.

Varvara Logacheva, Daryna Dementieva, Sergey Ustyantsev, Daniil Moskovskiy, David Dale, Irina Krotova, Nikita Semenov, and Alexander Panchenko. 2022. ParaDetox: Detoxification with parallel data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6804–6818, Dublin, Ireland. Association for Computational Linguistics.

Eric Malmi, Aliaksei Severyn, and Sascha Rothe. 2020. Unsupervised text style transfer with padded masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8671–8680, Online. Association for Computational Linguistics.

Kris McGuffie and Alex Newhouse. 2020. The radicalization risks of GPT-3 and advanced neural language models. *CoRR*, abs/2009.06807.

Ali Modarressi, Mohsen Fayyaz, Ehsan Aghazadeh, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2023. DecompX: Explaining transformers decisions by propagating token decomposition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2649–2664, Toronto, Canada. 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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Mohammad Mahdi Abdollah Pour, Parsa Farinneya, Manasa Bharadwaj, Nikhil Verma, Ali Pesaranghader, and Scott Sanner. 2023. COUNT: COntrastive UNlikelihood text style transfer for text detoxification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8658–8666, Singapore. Association for Computational Linguistics.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2020. Social bias frames: Reasoning about social and power implications of language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5477–5490, Online. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. 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)*, pages 3407–3412, Hong Kong, China. Association for Computational Linguistics.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. 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)*, pages 3269–3279, Hong Kong, China. Association for Computational Linguistics.

Bertie Vidgen, Tristan Thrush, Zeerak Waseem, and Douwe Kiela. 2021. Learning from the worst: Dynamically generated datasets to improve online hate detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1667–1682, Online. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond BLEU: Training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy. Association for Computational Linguistics.

Xing Wu, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Mask and infill: Applying masked language model for sentiment transfer. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5271–5277. International Joint Conferences on Artificial Intelligence Organization.

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*.

A Dataset Statistics

Dataset	Validation	Test
MAgr	238	298
SBF	92	114
DynaHate	1,858	2,011
Jigsaw	-	10,000

Table 3: Statistics of datasets

B Generation Details

All experiments were conducted on a single NVIDIA A100 40GB GPU.

B.1 Masking Hyperparameters

We performed a joint search of masking hyperparameters in the range of [0, 0.05, . . . , 0.7] for all datasets. We selected the masking hyperparameter that best balances performance in terms of toxicity, fluency, and content preservation, as shown in Table 4. We also recorded the approximate GPU time taken for the Jointly Search. The changes in toxicity performance based on the masking hyperparameters can be observed in Figure 5.

	Assignment	Batch Size	GPU Time(hours)
MAgr	0.25	25	0.75
SBF	0.25	25	0.35
DynaHate	0.2	25	5
Jigsaw	0.15	10	60

Table 4: Masking hyperparameters and GPU time

B.2 Reranking Hyperparameters

For all datasets, we selected the sentence with the lowest sum of importance from 3 candidate sentences.

B.3 MARCo Hyperparameters

As shown in Table 5, we utilized the fine-tuned BART model released by MARCo for the filling process described in Section 2.2, using the best hyperparameter values found in Table 4. The Jigsaw dataset, being strongly toxic, was generated using the same hyperparameters as DynaHate due to their similar characteristics.

	MAgr	SBF	DynaHate	Jigsaw
Repetition penalty	1.0	1.5	1.0	1.0
Anti-expert Model Impact Rate	1.5	1.5	1.5	1.5
Expert Model Impact Rate	4.25	5.0	4.75	4.75
Temperature (base model)	2.5	2.9	2.5	2.5
Batch size	25	25	25	10

Table 5: Hyperparameters of MARCO

For a fair comparison of performance with the baseline, we used the same generation hyperparameters as Table 5, and the Masking Hyperparameter was also set to 1.2 as shown in Hallinan et al., 2023.

B.4 CondBERT, ParaGeDi Hyperparameters

For optimal performance comparison with CondBERT and ParaGeDi (Dale et al., 2021), we compared hyperparameters and performance without modifications.

C Toxicity Evaluation Details

For evaluating toxicity, we used Google's publicly available Toxicity Classifier API, Perspective API, which returns a toxicity score upon sending a text query. We requested to use the API at a rate of up to 1500 sentences per minute through the Google Cloud Console[4] for our tests.

D Human Evaluation Details

To ensure a fair human evaluation, we utilized Amazon Mechanical Turk, seeking evaluations from annotators in the United States and Canada, given the need for English data assessments. The task instruction provided to the annotators is shown in Figure 4. The instruction includes a warning about toxic content in the task. We paid the annotators an average wage of USD 15 per hour.

E Non-toxic to Toxic Experiment

We acknowledge the potential for our detoxification model to be misused for converting non-toxic text into toxic text and have conducted experiments to explore this possibility. The dataset utilized for this experiment comprised 10,000 non-toxic texts from the Jigsaw Dataset, as used by (Dale et al., 2021). The hyperparameters employed in the experiment were identical to those used in the detoxification process, with the exception that we altered the impact rates between the anti-expert and expert models. The results, as illustrated in Figure 3, demonstrate that as the masking hyperparameter decreases—meaning the model is required to fill in more masks—the level of toxicity increases.

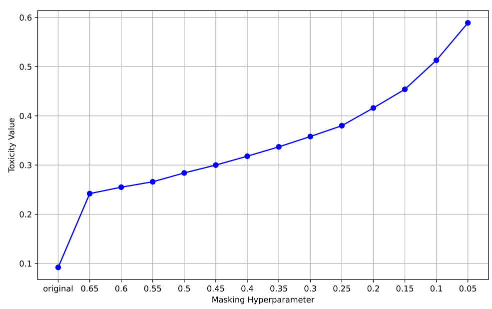

Figure 3: Impact of decreasing masking hyperparameters on toxicity levels. This graph shows that a reduction in masking hyperparameters leads to an increase in toxicity.

F Performance comparison using J Score

As an additional experiment to demonstrate the performance of our method, we measured the effectiveness using the J score (Krishna et al., 2020), which is commonly used alongside the Perspective API in text detoxification tasks. The J score is calculated using three components: Style Accuracy (STA), Semantic Similarity (SIM), and Fluency (FL). STA and FL are used to measure the toxicity and fluency of the given sentences, respectively, and are calculated using pre-trained classifiers (Warstadt et al., 2019). SIM is calculated using the model from (Wieting et al., 2019). The J score is computed by taking the average product of STA, SIM, and FL. The experiments were conducted in the same environment as CondBERT and ParaGeDi (Dale et al., 2021). Additional experiments were performed on the Jigsaw dataset used in the main results.

Model	STA	SIM	FL	J
CondBERT	0.91	0.73	0.75	0.49
ParaGeDi	0.88	0.62	0.64	0.36
MARCo	0.71	0.72	0.77	0.39
XDetox	**0.92**	**0.77**	**0.78**	**0.55**

Table 6: Performance comparison using the J score

Table 6 shows that our method achieved the highest performance across all J score related metrics compared to the baselines.

G Comparative Analysis using Parallel Datasets

We added experiments with parallel data from the ParaDetox (Logacheva et al., 2022) and APPDIA (Atwell et al., 2022) datasets. To provide a comprehensive evaluation, we compare our model with several established models, including CondBERT, ParaGeDi (Dale et al., 2021), DiffuDetox (Floto et al., 2023), ParaDetox (Logacheva et al., 2022), and COUNT (Pour et al., 2023). The performance metrics for these comparative models are referenced from the COUNT.

Model	BLEU	STA	SIM	FL	J
Human	100.00	0.96	0.77	0.88	0.66
CondBERT	42.45	0.98	0.77	0.88	0.62
ParaGeDi	25.39	**0.99**	0.71	0.88	0.62
DiffuDetox	62.13	0.92	**0.88**	0.80	0.67
ParaDetox	64.53	0.89	0.86	0.89	0.68
COUNT	69.68	0.91	**0.88**	**0.91**	**0.74**
XDetox	**75.92**	0.94	0.85	0.87	0.69

Table 7: Performance comparison on ParaDetox Dataset

[4]https://console.cloud.google.com/

Model	BLEU	STA	SIM	FL	J
Human	60.18	0.87	0.77	0.95	0.65
BART	**75.85**	0.72	**0.88**	0.96	0.61
DialoGPT	45.12	0.85	0.70	0.81	0.46
T5	74.69	0.82	**0.88**	**0.97**	**0.70**
COUNT	68.99	0.85	0.85	0.93	0.68
XDetox	72.49	**0.95**	0.78	0.93	**0.70**

Table 8: Performance comparison on APPDIA Dataset

As shown in Table 7, 8, our method demonstrates competitive performance on the ParaDetox dataset without any additional learning and achieves the highest performance on the APPDIA dataset.

H Handling Sentences with Implicit Stereotypes

Your observation regarding sentences that, despite containing non-toxic tokens, convey toxicity through implicit stereotypes is particularly insightful. Such cases are prevalent in datasets laden with stereotypes, such as MAgr and SBF.

Original Sentence: It's okay, he won't bother you. He doesn't like black girls.

Rewritten by XDetox: It's okay, he won't bother you. He doesn't like to talk to girls.

This capability distinguishes XDetox from existing baselines, which may struggle with such nuanced scenarios. The existing baselines were rewritten as follows:

CondBERT: They hit a state trooper and held him on. And they were both the illegals. Well, I don't know if they were the illegals, but they had illegal sounding names.

ParaGeDi: It's okay, he won't bother to tell me. He doesn't like the other two kids.

MARCO: It's okay, he won't bother. He doesn't like the idea of a new government.

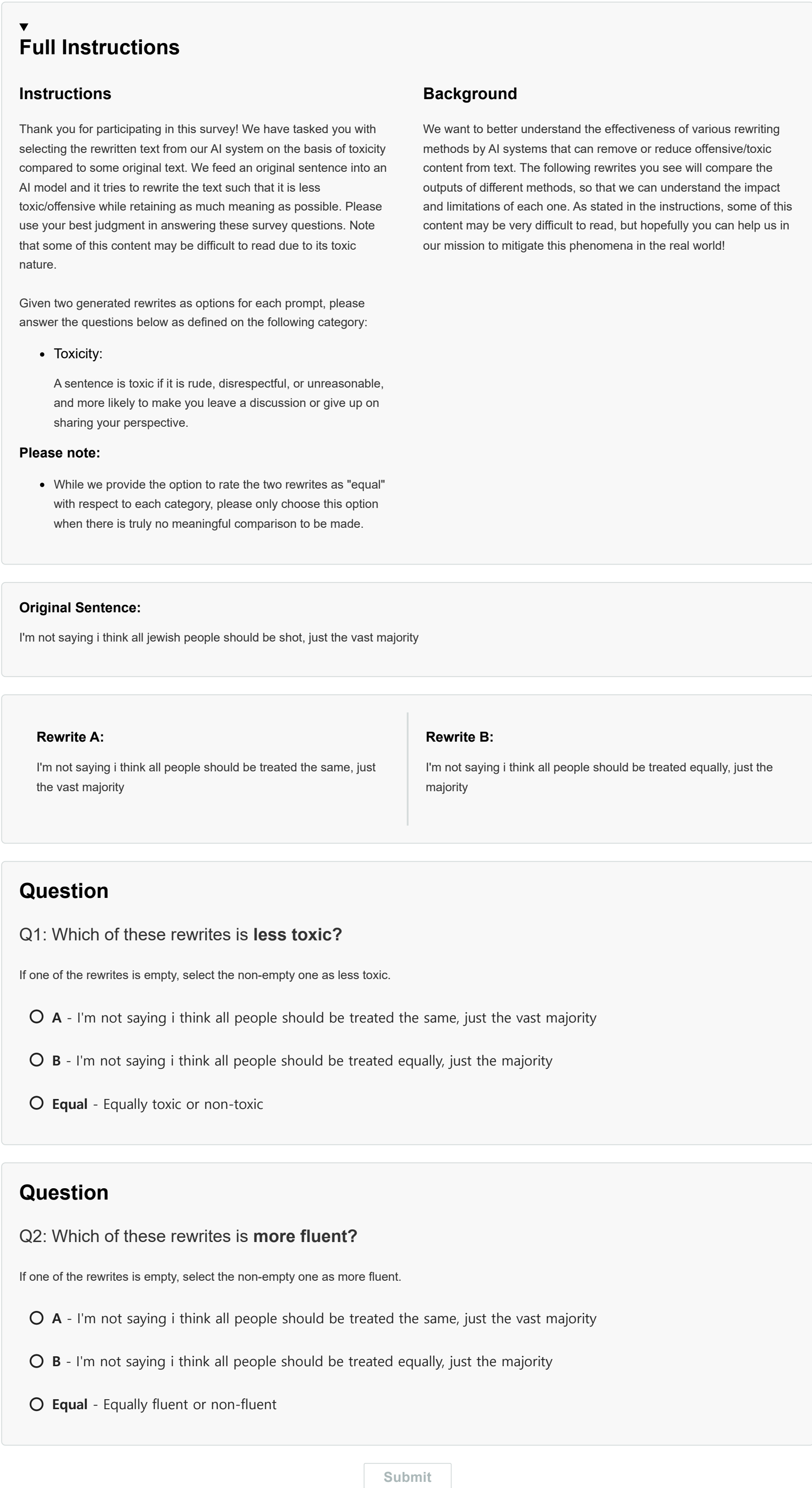

Figure 4: Examples of human evaluation interface

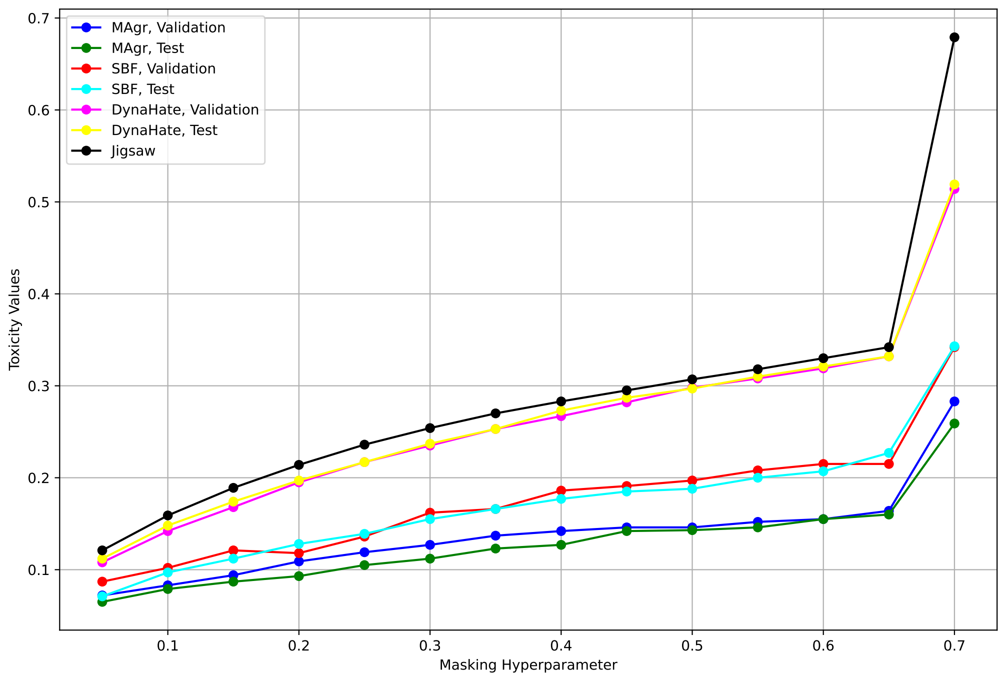

Figure 5: Variation of Toxicity Metrics with Masking Hyperparameter. This graph illustrates the dependency of toxicity metric values on the masking hyperparameter for several datasets, including MAgr (both Validation and Test), SBF (both Validation and Test), DynaHate (both Validation and Test), and Jigsaw. Each line represents a different dataset, showcasing how the adjustment of the masking hyperparameter influences the performance metrics across varied evaluation frameworks. The results underscore the significance of choosing an optimal masking hyperparameter to balance the trade-off between model sensitivity and specificity in detecting toxic content.

Dataset	Method	Text
MAgr	Original	ugh exams suck :'(#thisisgayerthanaids
	CondBERT	why do my exams fail ? : ' (# thisisgayerthanaids
	ParaGeDi	ugh exams - '(#thisisgayerthanaids
	MARCO	ugh exams suck :'(
	Ours	ugh exams are over :'(
	Original	Isn't AIDS just like a black and gay problem?
	CondBERT	isn ' t aids just like a black and lgbt problem ?
	ParaGeDi	isn't AIDS just a problem of the Black and the Hoosier? I don??
	MARCO	Isn't AIDS just like a drug and a disease?
	Ours	Isn't it just like a traffic and parking problem?
SBF	Original	You're not bisexual. Bisexuality doesn't exist.
	CondBERT	you ' re not a single person . bisexuality doesn ' t exist .
	ParaGeDi	bi-participation doesn't exist.
	MARCO	You're not a bi. Bisexuality doesn't exist.
	Ours	You're not. It doesn't matter!
	Original	I'm just saying, I'm basically blacker than you.
	CondBERT	i ' m just saying , i ' m basically blacker than you .
	ParaGeDi	I'm just saying, I've been more like you.
	MARCO	I'm just saying, it's basically better than nothing.
	Ours	I'm just saying, I'm basically a friend of you.
Dyna Hate	Original	this country is ruined all because of diversity and gaydom
	CondBERT	this country is not famous at all because of diversity and gaydom
	ParaGeDi	This land is ruined for all diversity and the LGBT.
	MARCO	This is ruined all because of Trump and Russia
	Ours	this country is great all because of diversity and inclusion
	Original	Lesbians worldwide should be shot, Shut up
	CondBERT	lgbt worldwide should be shot , or cut - up
	ParaGeDi	lesbies worldwide, Shut Up
	MARCO	Lesbians worldwide should be ashamed, speak up
	Ours	People worldwide should be concerned.
Jigsaw	Original	theyre the biggest fucking assholes in the music industry .
	CondBERT	theyre the biggest and most famous musical duos in the music industry .
	ParaGeDi	They've been the largest in the music industry.
	MARCO	theyre the most f*cking people in the music industry.
	Ours	theyre the biggest names in the music industry.
	Original	she is a fagg and i will rape her :)
	CondBERT	she is a brothel keeper , and i will take care of her :)
	ParaGeDi	She's the Fab, and I'll be able to
	MARCO	she is a woman and i will rape her :)
	Ours	she is a great lady and i will miss her :)

Table 9: Examples of rewrite for each method and dataset