Reflect, Reason, Rephrase (R^3-Detox): An In-Context Learning Approach to Text Detoxification

Anonymous ACL submission

Abstract

Traditional content moderation censors harmful content, which can often limit user participation. Text detoxification offers a better alternative, promoting civility without silencing voices. However, prior approaches oversimplify the task by treating detoxification as a one-step process, neglecting the deep contextual analysis needed to remove toxicity while preserving meaning. In this paper, we introduce R^3-Detox—a Reflect, Reason, and Rephrase framework that enhances detoxification through a structured three-step process, all executed within a single prompt. First, we instruct the LLM to analyze potential toxic words or phrases, guided by Shapley values from toxicity detectors, to counteract potential hallucinations. Next, the model assesses the overall toxicity of the sentence based on these identified elements. Finally, leveraging this prior analysis, the model reasons about necessary modifications to eliminate toxicity while maintaining meaning. We apply this framework and Self-Reflection models to enrich offensive content paraphrasing datasets—ParaDetox, Parallel Detoxification, and APPDIA—by adding explicit detoxification reasoning to each instance, which originally contained only input sentences and their paraphrases. We evaluate our methodology using In-Context Learning, comparing R^3-Detox against state-of-the-art methods on the same datasets. Experimental results show that our approach outperforms existing methodologies, even in instruction-following models.

Disclaimer: Figures and examples shown in this manuscript may feature toxic language.

1 Introduction

With the rapid spread of misinformation and hate speech on social media, scalable content moderation is essential to protect vulnerable groups (Maarouf et al., 2024; Arun et al., 2024). While traditional moderation methods, such as flagging and censoring harmful content,are effective (Gorwa et al., 2020; Davidson et al., 2017; Lees et al., 2022), they often restrict user participation and limit discussion diversity, highlighting the need for more advanced approaches.

Text detoxification offers a promising approach by rephrasing offensive content into less harmful language while preserving meaning (Logacheva et al., 2022). This fosters inclusive dialogue while addressing toxicity. However, its effectiveness depends on skilled annotators with deep contextual and societal understanding to ensure fairness and accurate interpretation.

Previous studies have explored the use of supervised generative models, such as BART (Logacheva et al., 2022) and DialoGPT (Atwell et al., 2022), for paraphrasing offensive content. While these models perform well on certain metrics, including BLEU, BERTScore, and ROUGE, they come with notable limitations. They require large amounts of labeled data, generalize poorly across domains, and often fail to fully eliminate toxic behavior (Som et al., 2024). To enhance adaptability, prior research has leveraged the In-Context Learning (ICL) (Zhou et al., 2024) capabilities of Large Language Models (LLMs), showing promising results in both detoxification (Som et al., 2024; He et al., 2024) and synthetic data generation (Moskovskiy et al., 2024). Additionally, recent approaches have leveraged the explanation capabilities of LLMs through Chain-of-Thought (CoT) prompting (Wei et al., 2022). This method asks the model to explain why a sentence is toxic before performing detoxification, yielding more interpretable and effective rewrites (Khondaker et al., 2024).

Expanding on recent advancements, we reconceptualize detoxification as a process of self-reflection (Li et al., 2023) and abductive reasoning. To this end, we introduce the **R^3-Detox** framework, which emulates human cognitive processes to enhance detoxification quality (Saldanha and

Kakas, 2020). This framework first guides the LLM in identifying potentially toxic words within a sentence using Shapley value-based explanations (Lundberg and Lee, 2017) extracted from toxicity detectors. Next, the framework instructs the LLM to analyze the underlying reasons for the sentence's toxicity based on these identified words. Finally, it directs the LLM to propose necessary modifications to neutralize the toxicity while preserving the original meaning, explaining how these changes promote a more inclusive and non-toxic output.

To ensure the quality of reasoning, we resort to models trained in Self-Reflection and incorporate existing detoxification datasets, conditioning on human-generated rephrasing to maintain consistency across each step. Our methodology addresses several key research questions central to evaluating the effectiveness of our framework:

- RQ1: How do existing reasoning evaluation metrics, correlate with human evaluation in the task of text detoxification?
- RQ2: Can models trained in Self-Reflection reason in highly subjective tasks, such as text toxicity detection and detoxification?
- RQ3: Does R^3-Detox achieve better detoxification results than state-of-the-art techniques by using ICL with few-shot examples?

Paper outline: Section 2 discusses related work, while Section 3 explains the methodology followed for the generation and validation of our proposed R^3-Detox framework. Section 4 outlines the evaluation metrics used and describes the human annotation process. Section 5 presents detailed experimental results and summarizes the key findings from the experiments. Finally, Section 7 summarizes the contributions and limitations of our study.

2 Related Work

Text style transfer (TST) aims to alter a sentence's style while preserving its meaning, with text detoxification focusing on converting toxic sentences into non-toxic ones (Jin et al., 2022).

Early detoxification approaches include supervised and unsupervised methods,to tackle this complex task and generate new synthetic data. Supervised methods like COUNT (Pour et al., 2023) introduce a contrastive unlikelihood objective, which maximizes the likelihood of generating non-toxic outputs and penalizes toxicity. Similar unsupervised approaches (Nogueira dos Santos et al., 2018; Laugier et al., 2021) address the lack of reference text while targeting toxicity.

Other approaches have modeled detoxification as a style-conditioned generation task, as in Dale et al. (2021), where they proposed CondBERT, a masked language detoxification methodology, and ParaGEDI, a controlled token-generation process.

To overcome generalization issues observed in previous methods, researchers have recently explored ICL for text detoxification, yielding superior outcomes compared to earlier approaches (Som et al., 2024; Moskovskiy et al., 2024). Some ICL methods use CoT reasoning to explain why a sentence is toxic before detoxification, and distill smaller models in the generated synthetic data, outperforming the baseline ICL method (Zhang et al., 2024; Khondaker et al., 2024).

Building on explanation-based approaches, we propose a method that not only explains why a sentence is toxic but also identifies and self-reflects on the modifications needed for detoxification. This is achieved through Self-Reflection models (Li et al., 2023). These models are trained using Self-Reflection tuning: an oracle LLM enhances the original training data by introspecting and improving the quality of instructions and responses.

3 Methodology

In this section we present and describe in detail the methodology outlined in Figure 1, pausing at each of its components: the datasets in use (Subsection 3.1), the few-shot example generation pipeline (Subsection 3.2), and the ICL method utilized to validate the R^3-Detox framework (Subsection 3.3).

3.1 Datasets

To validate our R^3-Detox framework through ICL with few-shot examples, we utilize public English text detoxification datasets to guide the reasoning process by using validated non-toxic paraphrases from these datasets. This ensures that the final reasoning regarding the changes needed to generate the non-toxic paraphrase is grounded on the difference between the original and paraphrased sentences. The detoxification datasets used are ParaDetox (Logacheva et al., 2022), APPDIA (Atwell et al., 2022), and Parallel Detoxification (Dementieva et al., 2021). Considering that the reflection step of of the R^3-Detox framework is responsi-

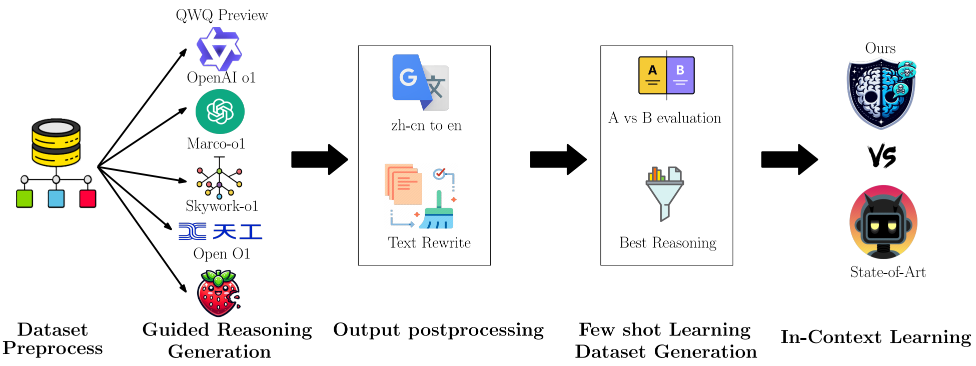

Figure 1: Overview of the methodology for the R^3-Detox framework. We first preprocess the datasets (Section 3.1) by extracting Shapley values from toxicity detectors. Guided reasoning is then generated using Self-Reflection models, ensuring no code-switching or data leakage so that the final non-toxic paraphrase is not explicitly present before detoxification. We evaluate models, select the best reasoning for each comment, and validate the few-shot examples by comparing them to state-of-the-art detoxification techniques using ICL.

ble for analyzing whether potentially toxic words, given a sentence, carry a toxic meaning, we introduce a non-toxic dataset to validate that it is indeed capable of differentiating the contextual meaning of toxic words within the sentence. Specifically, we use non-toxic comments from the Jigsaw Unintended Bias dataset (cjadams et al., 2019), selecting text samples with at least 10 annotators to ensure a sufficient sample size and guarantee that no toxic comments are included in the pool of non-toxic samples. In total, we collected 14,969 toxic sentences and 14,969 non-toxic sentences, creating a class-balanced dataset.

3.2 Generation of Few-Shot Examples

We introduce Shapley values from toxicity detectors to limit possible hallucinations during the reasoning process by incorporating prior knowledge about potentially toxic words. However, these toxicity models are not without issues, such as biases (Zhou et al., 2021), implicit toxicity (Hartvigsen et al., 2022), and generalization problems (Hanu and Unitary team, 2020). To address these challenges, we aggregate the Shapley values by selecting tokens identified as potentially toxic by models that agree on their toxicity. The models we use are specifically trained to mitigate these issues: Toxigen HateBERT and Toxigen RoBERTa from Hartvigsen et al. (2022), specialized in implicit toxicity detection; Toxic BERT and Unbiased Toxic RoBERTa from Detoxify, trained on vast amounts of data to overcome generalization issues; and ToxDetect RoBERTa Large from Zhou et al. (2021), trained to alleviate bias. Appendix A details further this process.

For the generation of abductive reasoning, we use several open-source models: Marco-o1 (Zhao et al., 2024), QwQ Preview (Team, 2024b), OpenO1 LLaMA 8B v0.1 (OpenSource-O1, 2024), and Skywork-o1-Open-Llama-3.1-8B (o1 Team, 2024), as well as the private OpenAI o1 (OpenAI et al., 2024). The reasoning generation is guided by the Shapley values and constraint by the non-toxic paraphrases in the dataset by providing the possible toxic words and the final non-toxic paraphrase in the prompt as context.

While the models generate high-quality reasoning, we encountered several issues along the way, including code-switching, a tendency to restate the provided non-toxic paraphrase, and instances where the OpenAI moderation tool flagged our queries as toxic. To overcome these problems, we employ the Qwen 2.5 32B model (Team, 2024a) to identify and eliminate unwanted behaviors, resort to the Google Translate API to mitigate language-mixing problems, and apply the latest jail-breaking technique introduced in Hughes et al. (2024). Further details of this process and the corresponding prompts are given in Appendix A.

Finally, to construct the dataset, we select the best reasoning outputs based on the JudgeLLM (Zhu et al., 2025) evaluation model, which is later described in Section 4. JudgeLLM has the highest correlation with human annotations, as is later empirically shown in Section 6.1. We further refine our selection by leveraging the best-performing model identified from our experiments made in response to RQ2 (Section 6.2). The final reasoning outputs is determined by an “A vs. B” tournament-style evaluation (detailed in Section 4.1).

3.3 In-Context Learning

ICL is an approach that consists of three components: 1) an instruction I explaining the task to be performed, 2) a set of n demonstrations from the Reflect, Reason, Paraphrase generated dataset, and 3) a query, which is the toxic sentence that needs to be rewritten. In our framework, we adopt the methodology recently proposed in Som et al. (2024), which selects the most similar sentences based on a content similarity model, all-mpnet-base-v2. The prompts used in our framework are presented in Appendix A.

4 Evaluation Framework

In this section, we present the metrics used to evaluate the generated reasoning and the non-toxic paraphrases. For each task, we describe the metrics applied and the human evaluation procedure. For all human evaluation procedures, we conducted annotations with three volunteers (two females and one male) aged 25 to 31 from Western Europe. The evaluation process is detailed in Appendix B.

4.1 Reasoning Evaluation

For the evaluation of the reasoning quality, we employ the ROSCOE metric suite Golovneva et al. (2022), which includes various sub-metrics: ROSCOE-SA (semantic alignment), ROSCOE-SS (semantic similarity), ROSCOE-LC (English grammatical acceptability scored by a classifier model), Discourse Representation (contradiction probability for each reasoning step) and Coherence (maximum contradiction probability between each reasoning step and previous steps).

In addition, we leverage LLMs for evaluation, particularly the JudgeLLM model. JudgeLLM, built upon Vicuna, was trained on a large-scale dataset of LLM-generated responses across diverse Natural Language Generation tasks, incorporating judgments from GPT-4. It achieves an agreement rate exceeding 90% in certain tasks (Zhu et al., 2025). JudgeLLM supports multiple evaluation methods, either referencing a gold standard or directly comparing multiple responses.

To assess the alignment of our proposed metrics with human preferences, we introduce an “A vs. B” comparative framework to rank models based on their performance in R^3-Detox reasoning. For each non-toxic paraphrase, we evaluate the reasoning generated by n models by systematically comparing each model against every other in a paired tournament. To determine the overall ranking, each pairwise comparison awards 1 point to the superior model, while ties or losses result in 0 points for the tied counterparts and the inferior model, respectively. The total number of tournaments is given by $\binom{n}{2} \cdot h = \frac{n!}{2!(n-2)!} \cdot h$, where h denotes the number of non-toxic paraphrases for reasoning evaluation.

In our experiments, considering the cost constraints of human evaluation, we assess a total of 20 instances, with 5 instances per dataset. This results in $\binom{5}{2} \cdot 20 = 200$ tournaments. Each tournament is evaluated by three volunteer annotators, yielding an inter-annotator agreement (measured as Fleiss' Kappa) of 0.183. If no clear winner is selected due to each annotator either choosing “win A”, “tie”, or “win B”, we classify it as a tie.

4.2 Paraphrase evaluation

To evaluate the generated non-toxic paraphrases, we use several traditional metrics from prior works (Logacheva et al., 2022; Khondaker et al., 2024): Style Transfer Accuracy (STA) (Logacheva et al., 2022), BERTScore (Gao et al., 2021), Content Preservation (SIM) (Wieting et al., 2019), Fluency (FL) (Warstadt et al., 2019), Joint Score (J) (Logacheva et al., 2022), and Toxicity Score (Tox) (Som et al., 2024). A more detailed explanation of each metric can be found in Appendix B.

Additionally, as proposed for reasoning evaluation, we use JudgeLLM to assess the quality of the generated paraphrases. To evaluate the alignment of the proposed metrics with human preferences, we introduce a triplet elimination tournament ranking method $m \cdot h = \frac{n-1}{2} \cdot h$, where m represents the number of triplet evaluations per toxic comment. In this evaluation method, models are randomly grouped into triplets, and each group competes against itself. The winner of each group progresses to the next phase, with the process repeating until only one model remains. This approach requires multiples of three participants. With this method we reduce annotation overhead while ensuring a more realistic evaluation of the subtle differences among the generated paraphrases and their comparison to the original sentence. To generate the final score, Borda count is used to aggregate the rankings of all the data points.

In our experiments, we evaluate a total of 51 toxic comments, with 17 comments selected from each of the APPDIA, Paradetox, and Parallel Detoxification datasets. The evaluation follows a triplet elimination tournament format, carried out in two distinct phases. In this setup, we compare 5 models using our proposed approach against 4 state-of-the-art techniques, as detailed in Section 5. In the first phase, the 3 annotators independently annotate 153 common triplets. In the second phase, based on the results of the first phase, each annotator evaluate their last 51 triplets, where the winners of the last phase face each other. The inter-annotator agreement from the first phase is calculated using Fleiss' Kappa, which results in a value of 0.09, indicating slight agreement. This relatively low agreement highlights the inherent subjectivity involve in evaluating the generated paraphrases. A more in-depth discussion of these complexities is presented in the Section 7, with detailed examples provided in Appendix C.

5 Experimental Setup

We now present the experimental setup for the generation of the few-shot examples (Subsection 5.1) and explain the ICL methodology performed to evaluate the R^3-Detox framework (Subsection 5.2).

5.1 Few-Shot Synthetic Data Generation

To validate the capabilities of the Self-Reflection models presented in Section 3.2 when used in the proposed R^3-Detox framework, we adopt the "A vs. B" evaluation method outlined in Section 4.1, applying it to 20 comments manually analyzed by three annotators. Additionally, we perform the same analysis using the following metrics: ROSCOE-SA, ROSCOE-SS, ROSCOE-LC, discourse representation, coherence, and the JudgeLLM model, all within the context of the same tournament ranking framework. Finally, we compute the correlation between the metrics, LLM evaluations, and human annotations by applying Spearman's Rank Correlation.

5.2 In-Context Learning

We perform ICL using five open-source models: Marco-o1, OpenO1, and QwQ Preview, selected based on their strong performance in the reasoning generation, as evaluated in Section 6.2. Additionally, we include the Llama 3.1 8B and Qwen 2 7B instruct models to compare them against OpenO1 and Marco-o1, respectively. Marco-o1 and OpenO1 were fine-tuned on the self-reflection task derived from these instruction models.

For our experiments, we select the following numbers of examples: [0,1,2,3,5,7,10]. This selection is constrained by the maximum context length

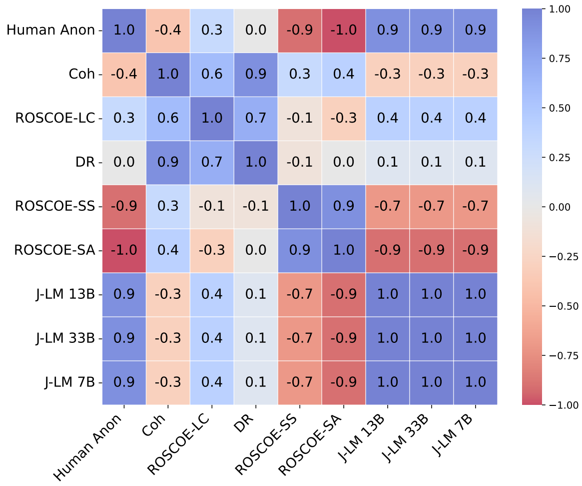

Figure 2: Spearman's rank correlation coefficients among metrics. J-LM means JudgeLM, DR is Discourse Representation, and Coh denotes Coherence.

of 8,096 tokens due to the limited computational resources available for the study. To validate our approach, we compare our results against human-annotated non-toxic paraphrases from each dataset, as well as several baseline methods: DetoxLLM from Khondaker et al. (2024), the BART model trained for the detoxification task by ParaDetox (Logacheva et al., 2022), and the ICL method introduced in PseudoParaDetox (Moskovskiy et al., 2024) for synthetic data generation, which utilized the dolphin-2.9-llama3-8b ablated model. We also considere the first ICL method proposed recently for detoxification in CAPP (Som et al., 2024).

We use the metrics listed in Section 4.2 to compare the performance of each ICL model on the same dataset. However, for the last approach (CAPP), a direct comparison is not possible, as GPT-3.5 models are no longer available and no code is provided. Instead, we use the example-generated dataset available in their repository[1].

6 Results and Discussion

This section presents the results for each of the research questions posed in Section 1. Code and dataset are available at: R-3-Detox (MIT license).

6.1 RQ1: Correlation of Reasoning Evaluation Metrics and Human Annotated Rankings

Figure 2 illustrates Spearman's rank correlation (ρ) among all metrics, including the majority vote

[1]CAPP article GitHub: `https://github.com/anirudhsom/CAPP-Dataset`, accessed on 02/13/2025.

of the human annotations introduced in Section 4.1. All rankings are computed using the "A vs B" pairwise comparison described in Section 4.1, calculated across the 20 comments in an overall 200 pair tournament. The correlation matrix depicts the correlation of the ranks assigned to the aggregated ranking of the pairwise tournaments.

The figure shows a strong correlation between JudgeLLM models and the human annotation majority vote, with the three variants achieving a ρ of 0.90. Although each model shows a similar Spearman rank correlation, when examining the aggregated Spearman rank correlation at the instance level, we observe that we obtain aggregated Spearman rankings using Fisher's method of 0.75, 0.736, and 0.769, with p-values of 0.08, 0.02, and 0.01 for JudgeLLM 7B, 13B, and 33B, respectively. These granular aggregated values show that only the JudgeLLM 33B and 13B models demonstrate a statistically significant difference from the null hypothesis, as indicated by their p-values of 0.01 and 0.02. On the other hand, the JudgeLLM 7B model, with a p-value of 0.08, does not show significant deviation from randomness, highlighting the variability in performance based on model size and complexity. Overall, the JudgeLLM 33B shows the best correlation with human annotators.

In contrast, the ROSCOE metrics correlate very poorly, with the best among the metrics being ROSCOE-LC, which has a ρ of 0.3. These results are expected, as these metrics only account for semantic consistency, logicality, informativeness, fluency, and factuality of the generated reasoning, rather than its content itself.

Rank	Human	*Score*	JudgeLM 33B	*Score*
1	QwQ Preview	43	QwQ Preview	57
2	OpenO1	42	Marco-o1	50
3	Marco-o1	39	OpenO1	49
4	OpenAI o1	27	OpenAI o1	25
5	Skywork-o1	2	Skywork-o1	5

Table 1: Human and JudgeLLM rankings, including the final scores obtained from the pairwise comparisons.

To end with the correlation analysis in response to RQ1, Table 1 presents a comparison of the final rankings based on the scoring methodology explained in Section 4.1 between the manual and automatic pairwise comparisons performed. As we can observe in this table, the overall ranking is practically the same across models. A exception are the rankings of Marco-o1 and OpenO1—models which, in both cases, score practically identical values relative to each other. By examining the obtained scores, we can observe that JudgeLLM tends to be more extreme, generating ties for 24 out of 200 pairwise comparisons, whereas the human evaluation is more lenient, generating ties in 47 out of 200 comparisons. Although we observe a disparity in the scores in the top part of the ranking, the lower ranks are similar in both cases. Regarding Skywork-o1, the poor results reflect the tendency of this model to generate code instead of resolving the task with a plain text output. The OpenAI-o1 model perform worse because they only provide the final result, lacking intermediate reasoning. Additionally, bypassing the OpenAI Moderation tool and using the prompt injection technique (Hughes et al., 2024) introduce noise, degrading the quality of the produced output.

6.2 RQ2: Can Self-Reflection models Reflect, Reason and Rephrase?

During the annotation process, we evaluate the correlation of the metrics, including those based on LLM, as shown in Figure 2. Most Self-Reflection models successfully identify toxic words in a sentence, assess overall toxicity, and suggest necessary changes. For non-toxic sentences, they explain why no toxicity is present, analyzing why potential toxic words are not harmful. Appendix C provides examples of their reasoning process.

Given that Marco-o1, OpenO1, and QwQ Preview perform best in the human annotation process, we further differentiate their capabilities by generating reasoning outputs for the entire dataset. To validate results, we apply the same "A vs. B" tournament from Section 4.1, using JudgeLLM 33B due to its high correlation with human evaluations.

JudgeLLM performs 89,856 pairwise evaluations on 29,952 dataset samples. OpenO1 ranks highest, winning 32,518 out of 86,526 possible scores, followed by QwQ Preview with 27,104 and Marco-o1 with 23,687. However, since the top model wins only 39.03% of the total pairs, all three models demonstrate strong performance in the Reflect, Reason, and Paraphrase reasoning tasks. The only limitation is the code-switching behavior observed in Marco-o1 and QwQ Preview.

6.3 RQ3: Reflect, Reason, Rephrase ICL vs State-of-Art

In this section, we first analyze the impact of the number of demonstrations on the results (Section 6.3.1). Then, in Section 6.3.2, we compare our

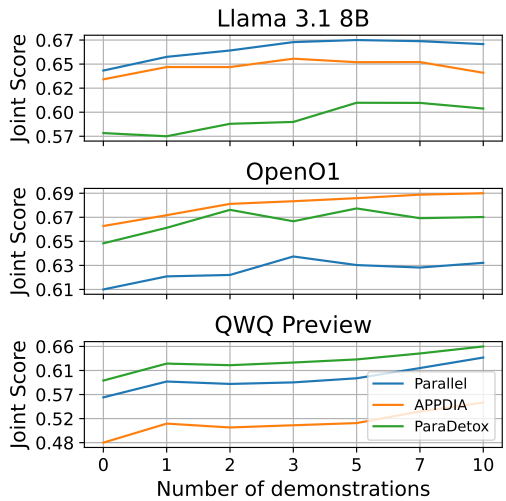

Figure 3: Joint Score versus the number of examples.

approach with other state-of-the-art techniques. Finally, in Section 6.3.3, we examine the correlation between the metrics introduced in Section 4.2.

6.3.1 Importance of Number of Demonstrations

Figure 3 shows the impact of introducing demonstrations on the Joint Score metric across a set of models to highlight the variability of this impact (we refer to Appendix C for other metrics and models). We observe that after two examples are provided, the Joint Score generally improves from the zero setting, following the Reflect, Reason, Rephrase reasoning style. Furthermore, while increasing the number of examples improves performance, the effect varies depending on the model. For example, the QwQ Preview model tends to improve as more demonstrations are provided, contrarily to Llama 3.1 8B and OpenO1. As more sentences are provided, the output of these latter models become less similar to the query, potentially introducing noise into the model's performance.

Notably, models fine-tuned in Self-Reflection, such as OpenO1 derived from the base Llama 3.1 8B model, appear to be more capable, adjusting better to the task of detoxification compared to the base instruction-tuned models.

6.3.2 Comparison with other approaches

The comparison between R^3-Detox and state-of-the-art models is presented in Table 2. The objective of any paraphraser is to achieve high content similarity and generation quality while maintaining a low toxicity score in the generated paraphrase. In this context, we analyze the content similarity results for the APPDIA, ParaDetox, and Parallel Detoxification datasets.

In this table we observe that the BART and PseudoParaDetox approaches yield the best BERT-F1, BLEU, and SIM scores. However, although these approaches seem to be promising based on these metrics, their toxicity and STA scores indicate that they maintain higher toxicity than the Gold Standard. Regarding FL, we find that all approaches, except BART, produce similarly high-quality phrases, outperforming the Gold Standard. This suggests that the models tend to correct typographical errors in the original sentences. Upon analyzing the toxicity levels (STA and Tox), it is evident that our approach and DetoxLLM achieve the best results, indicating that incorporating prior reasoning into the paraphrasing process significantly improves detoxification. Finally, when considering the Joint Score, our approach generally outperforms the others, achieving better overall detoxification quality. The only model that deviates from the expected standard is QwQ Preview, a highly capable model that achieves the best detoxification but deviates too much from the original content, as reflected in the content similarity results. This phenomenon can be attributed to the tendency of this reasoning model to overthink or over-contextualize the sentence, as further discussed in Section 7.

Overall, our R^3-Detox approach generates paraphrases that are highly similar in meaning while maintaining a low toxicity score, closely followed by DetoxLLM. This emphasizes the importance of incorporating a prior reasoning process to ensure better detoxification.

6.3.3 Correlation Between Metric Rankings and Annotator Evaluations

Figure 4 shows the correlation between the rankings of the metrics (Section 4.2) and those of the three annotators, given their low annotation agreement. The Spearman's rank correlation between annotators is weak (<0.2), with p-values of 0.09, 0.27 and 0.18 when comparing Annotators 1 and 2, 1 and 3, and 2 and 3, respectively. Additionally, all p-values reported in Appendix C are above 0.05, suggesting a weak agreement and concluding that the annotators' rankings are not highly consistent.

7 Conclusions

In this paper, we introduce a new framework, coined *Reflect, Reason, and Rephrase* (R^3-Detox), which transforms the text detoxification task into a

Dataset	Method	BERT-F1 ↑	BLEU ↑	SIM ↑	FL ↑	STA ↑	J ↑	Tox ↓
APPDIA	Original Sentence	-	-	-	-	-	-	0.748
	Gold-Standard	0.954	0.516	0.784	0.912	0.887	0.634	0.134
	BART	0.972	0.668	0.881	0.861	0.808	0.612	0.221
	DetoxLLM	0.925	0.214	0.654	0.967	0.922	0.583	0.059
	PseudoParaDetox	0.95	0.442	0.772	0.949	0.778	0.57	0.203
	CAPP*	0.955	0.521	0.808	0.971	0.898	0.704	0.117
	(R^3-Detox) Marco-o1	0.936 ± 0.002	0.336 ± 0.013	0.692 ± 0.012	0.948 ± 0.007	0.925 ± 0.009	0.607 ± 0.01	0.077 ± 0.01
	(R^3-Detox) Qwen 2.5 7B	0.926 ± 0.011	0.284 ± 0.068	0.649 ± 0.065	0.932 ± 0.026	0.958 ± 0.012	0.577 ± 0.04	0.048 ± 0.015
	(R^3-Detox) OpenO1	0.934 ± 0.003	0.324 ± 0.015	0.686 ± 0.016	0.963 ± 0.006	0.948 ± 0.008	0.627 ± 0.008	0.055 ± 0.007
	(R^3-Detox) Llama 3.1 8B	0.93 ± 0.005	0.326 ± 0.035	0.653 ± 0.031	0.945 ± 0.015	0.959 ± 0.01	0.593 ± 0.014	0.053 ± 0.01
	(R^3-Detox) QwQ Preview	0.909 ± 0.004	0.183 ± 0.022	0.529 ± 0.027	0.987 ± 0.002	0.986 ± 0.004	0.515 ± 0.024	0.02 ± 0.004
ParaDetox	Original Sentence	-	-	-	-	-	-	0.892
	Gold-Standard	0.951	0.47	0.813	0.805	0.943	0.617	0.0763
	BART	0.961	0.555	0.862	0.831	0.924	0.662	0.091
	DetoxLLM	0.922	0.203	0.68	0.967	0.951	0.625	0.033
	PseudoParaDetox	0.943	0.394	0.799	0.923	0.859	0.633	0.117
	CAPP*	0.955	0.486	0.849	0.939	0.945	0.754	0.06
	(R^3-Detox) Marco-o1	0.94 ± 0.002	0.366 ± 0.012	0.771 ± 0.008	0.904 ± 0.002	0.936 ± 0.006	0.652 ± 0.009	0.064 ± 0.003
	(R^3-Detox) Qwen 2.5 7B	0.931 ± 0.011	0.32 ± 0.072	0.734 ± 0.066	0.903 ± 0.033	0.969 ± 0.009	0.641 ± 0.033	0.036 ± 0.012
	(R^3-Detox) OpenO1	0.938 ± 0.002	0.349 ± 0.016	0.767 ± 0.014	0.931 ± 0.007	0.947 ± 0.003	0.677 ± 0.009	0.054 ± 0.001
	(R^3-Detox) Llama 3.1 8B	0.936 ± 0.005	0.358 ± 0.037	0.747 ± 0.03	0.919 ± 0.02	0.967 ± 0.003	0.663 ± 0.011	0.043 ± 0.005
	(R^3-Detox) QwQ Preview	0.921 ± 0.005	0.247 ± 0.026	0.663 ± 0.029	0.972 ± 0.006	0.979 ± 0.006	0.631 ± 0.02	0.025 ± 0.004
Parallel	Original Sentence	-	-	-	-	-	-	0.836
	Gold-Standard	0.934	0.369	0.724	0.801	0.92	0.533	0.09
	BART	0.966	0.63	0.875	0.876	0.794	0.609	0.165
	DetoxLLM	0.922	0.203	0.68	0.967	0.951	0.625	0.033
	PseudoParaDetox	0.946	0.423	0.807	0.929	0.781	0.585	0.163
	(R^3-Detox) Marco-o1	0.937 ± 0.002	0.368 ± 0.014	0.756 ± 0.011	0.927 ± 0.005	0.917 ± 0.005	0.643 ± 0.008	0.074 ± 0.009
	(R^3-Detox) Qwen 2.5 7B	0.928 ± 0.011	0.315 ± 0.073	0.711 ± 0.067	0.921 ± 0.028	0.955 ± 0.014	0.624 ± 0.036	0.044 ± 0.014
	(R^3-Detox) OpenO1	0.934 ± 0.002	0.343 ± 0.017	0.746 ± 0.015	0.953 ± 0.01	0.935 ± 0.006	0.665 ± 0.009	0.061 ± 0.004
	(R^3-Detox) Llama 3.1 8B	0.933 ± 0.005	0.359 ± 0.038	0.725 ± 0.026	0.937 ± 0.021	0.95 ± 0.011	0.646 ± 0.007	0.056 ± 0.007
	(R^3-Detox) QwQ Preview	0.916 ± 0.005	0.236 ± 0.026	0.627 ± 0.031	0.981 ± 0.003	0.974 ± 0.008	0.599 ± 0.024	0.028 ± 0.007

Table 2: Quantitative assessment of different LLMs based on our R^3-Detox approach and comparison against state-of-the-art detoxification techniques is presented. The toxicity of the original sentence is provided, and the dataset's non-toxic paraphrase metric is used as the Gold Standard. In our approach, the mean and standard deviation (std) values are computed over the different few-shot values. Best and worst results are shaded in blue and red, respectively. CAPP* values are based on the small subset available in their GitHub repository (Som et al., 2024).

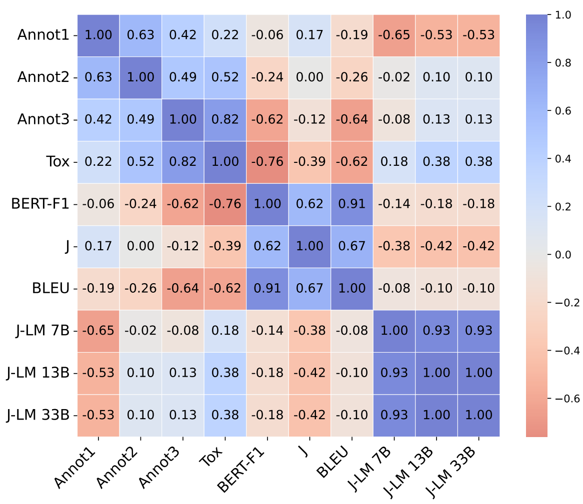

Figure 4: Spearman's rank correlation coefficients among metrics used to evaluate detoxification results.

three-step reasoning process. Through this methodology, we generate the first dataset that explains the intermediate analysis required to produce a non-toxic final sentence using open-source Self-Reflection models. To validate the quality of the generated reasoning, we use human evaluation to assess how well human annotations correlate with the ROSCOE metric and JudgeLLM evaluations. Based on this correlation, we select the most suitable metric for evaluating the intermediate analysis generated by the Self-Reflection models, allowing us to determine the best-performing models. This dataset is later used as a few-shot example set in ICL to validate our R^3-Detox framework in generating non-toxic paraphrases from toxic inputs by explaining why a given sentence is toxic and what changes are needed to make it non-toxic.

In the experiments discussed in this study, we have observed that the JudgeLLM evaluation model exhibits a high correlation with human annotation, making it a suitable tool for assessing the generated reasoning. We have also demonstrated that R^3-Detox, based on the generated demonstrations, outperforms existing state-of-the-art techniques, producing rephrased sentences that are less toxic and retain their original meaning. Finally, we have explored the manual annotation of the generated paraphrases, concluding that it is a complex, subjective task that requires standardized guidelines to improve the consistency of annotations.

Limitations

Although our R^3-Detox framework offers meaningful contributions to detoxification, certain limitations must be acknowledged to provide a comprehensive understanding of our findings.

To begin with, the Reflect, Reason, and Rephrase methodology has certain limitations regarding its underlying components: Shapley values extracted from toxicity classifiers and Self-Reflection models. Concerning the extracted explanations from toxicity detectors, even though we have used models trained to mitigate biases, improve the capacity to detect implicit toxicity, and enhance generalization to other domains, these issues are not completely resolved. This leads to cases where, for example, the mere presence of the word "gay" increases the probability of a sentence being flagged as toxic. This can be observed in the Shapley values extracted from ToxDetect RoBERTa for the few-shot example generation. To address such limitations, we use these explanations as potential toxic words to inform the model that they might not always be accurate. However, this remains an existing limitation of LLMs, which the research community is actively working to address (Gallegos et al., 2024).

Regarding the models trained on Self-Reflection, due to their introspective training, we have observed cases–especially with the QwQ Preview model, as outlined in Section 6.3.2–where the model over-contextualizes the societal background or context in which a sentence may be formulated. This issue directly affects the generated paraphrases, leading to discrepancies in meaning. For example, the sentence "A sociopathic idiot is trying to reassure us" is paraphrased as "Given their history of unreliable behavior, their attempts to reassure us are met with skepticism," which significantly alters the original meaning.

An additional area for improvement is the evaluation setting and process used for assessing the generated paraphrases. In this study, we have worked with a limited evaluation dataset, which should be expanded to better assess the framework's capabilities in more complex environments, such as implicit toxicity comments, contextualized toxicity in conversation settings, and non-detoxifiable cases:

- **Implicit toxic comments:** These contain harmful language that is indirect, subtle, or disguised—often relying on sarcasm, coded language, stereotypes, or dark humor rooted in offensive remarks. They are particularly challenging for classical classifiers to detect, as they are scarce in available datasets (Hartvigsen et al., 2022) and cannot be easily collected using heuristic rules. Further research is needed to assess the effectiveness of existing approaches in these challenging settings, where accurately interpreting text meaning and intent is crucial for distinguishing toxicity from benign language.

- **Detoxification in conversational contexts:** As text detoxification shifts from isolated comments to full conversations, ensuring that the framework maintains the intended meaning without misinterpreting the broader conversational context becomes a challenge. Misinterpretations could result in paraphrases that alter the original intent or fail to appropriately mitigate toxicity.

- **Non-detoxifiable toxic comments:** Some toxic comments are too offensive to be rephrased without completely changing their meaning (Khondaker et al., 2024). In this study, we do not address extreme cases of non-detoxifiable toxic comments, as widely used benchmark datasets for text detoxification, such as APPDIA, ParaDetox, and Parallel Detoxification, do not include them. Consequently, our approach has not been evaluated in such scenarios. However, we recognize the importance of developing strategies to effectively handle these challenging cases.

Additionally, we note that text detoxification techniques may introduce potential risks, as the generated outputs can still contain subtle forms of toxicity or rudeness that are difficult to detect using classical classifiers. Such subtle negative comments could be exploited for malicious purposes, potentially bypassing moderation tools on platforms while concealing an underlying offensive meaning.

Our final observed limitation is the subjectivity involved in annotating generated paraphrased comments. The perception of toxicity can vary between annotators due to cultural, personal, or contextual differences, making it challenging to achieve consistent evaluations, as noted in Section 6.3.3. These challenges can be summarized as follows:

- **Synonym variability**: Models may generate sentences that are identical except for synonymous words, which can lead to inconsistencies if annotators perceive one synonym as more appropriate or less offensive than another. For example, "She's kind of not very bright too." versus "She's kind of not very smart too." when paraphrasing "She's kind of stupid too."

- **Ambiguity in meaning preservation**: Defining what maintaining the same meaning refers to in a rephrasing task is inherently difficult. Our approach prioritizes minimal modifications to retain the original intent, but this can result in paraphrases that remain subtly rude due to lingering connotations. For instance, the original toxic sentence "I hope the bastard suffered" and its paraphrases: "I hope the person had a difficult time." or "I hope the person suffered." still convey varying degrees of negativity.

- **Challenges with slang and highly specific terms**: Some sentences include niche terminology or slang (e.g., "sandngr"), making it difficult for annotators or LLMs to accurately interpret their meaning. This adds an additional layer of subjectivity to the evaluation process, as misunderstandings can impact the consistency of toxicity assessments.

In summary, these limitations highlight the persistent challenges in addressing toxicity through paraphrasing. The diverse ways in which toxicity manifests, the complexities of maintaining consistent annotations, and the risk of subtle toxic outputs that go undetected all pose major obstacles. Additionally, the reliance on existing datasets may limit the framework's ability to handle extreme cases of toxicity, implicit toxic language, or nuanced conversational contexts. Furthermore, the subjectivity involved in evaluating paraphrases complicates the development of standardized assessment criteria. Addressing these challenges requires continued refinement of paraphrasing models, the integration of diverse datasets, and the development of more robust evaluation methodologies.

Acknowledgments

We acknowledge the use of AI assistants for code assistance and writing style enhancement. Acknowledgments to research grants are omitted in compliance with the double-blind review policy.

References

Arvindh Arun, Saurav Chhatani, Jisun An, and Ponnurangam Kumaraguru. 2024. X-posing free speech: Examining the impact of moderation relaxation on online social networks. In Proceedings of the 8th Workshop on Online Abuse and Harms (WOAH 2024), pages 201–211, Mexico City, Mexico. Association for Computational Linguistics.

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, Daniel Borkan, inversion, Jeffrey Sorensen, Lucas Dixon, Lucy Vasserman, and Nithum Thain. 2019. Jigsaw unintended bias in toxicity classification.

David Dale, Anton Voronov, Daryna Dementieva, Varvara Logacheva, and et al. 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.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In Proceedings of the 11th International AAAI Conference on Web and Social Media, ICWSM '17, pages 512–515.

Daryna Dementieva, Sergey Ustyantsev, David Dale, Olga Kozlova, Nikita Semenov, Alexander Panchenko, and Varvara Logacheva. 2021. Crowdsourcing of parallel corpora: the case of style transfer for detoxification. In CSW@VLDB.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, and et al. 2024. Bias and fairness in large language models: A survey. Computational Linguistics, 50(3):1097–1179.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. CoRR, abs/2104.08821.

Olga Golovneva, Moya Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2022. Roscoe: A suite of metrics for scoring step-by-step reasoning. CoRR, abs/2212.07919.

Robert Gorwa, Reuben Binns, and Christian Katzenbach. 2020. Algorithmic content moderation: Technical and political challenges in the automation of platform governance. Big Data & Society, 7(1):2053951719897945.

Laura Hanu and Unitary team. 2020. Detoxify. Github. https://github.com/unitaryai/detoxify.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, and et al. 2022. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.

Xinlei He, Savvas Zannettou, Yun Shen, and Yang Zhang. 2024. You Only Prompt Once: On the Capabilities of Prompt Learning on Large Language Models to Tackle Toxic Content . In 2024 IEEE Symposium on Security and Privacy (SP), pages 770–787, Los Alamitos, CA, USA. IEEE Computer Society.

John Hughes, Sara Price, Aengus Lynch, Rylan Schaeffer, Fazl Barez, and et al. 2024. Best-of-n jailbreaking. Preprint, arXiv:2412.03556.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2022. Deep learning for text style transfer: A survey. Computational Linguistics, 48(1):155–205.

Md Tawkat Islam Khondaker, Muhammad Abdul-Mageed, and Laks V. S. Lakshmanan. 2024. DetoxLLM: A framework for detoxification with explanations. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 19112–19139, Miami, Florida, USA. Association for Computational Linguistics.

Léo Laugier, John Pavlopoulos, Jeffrey Sorensen, and Lucas Dixon. 2021. Civil rephrases of toxic texts with self-supervised transformers. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 1442–1461, Online. Association for Computational Linguistics.

Alyssa Lees, Vinh Q. Tran, Yi Tay, Jeffrey Sorensen, Jai Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective api: Efficient multilingual character-level transformers. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '22, page 3197–3207, New York, NY, USA. Association for Computing Machinery.

Ming Li, Lichang Chen, Jiuhai Chen, Shwai He, and Tianyi Zhou. 2023. Reflection-tuning: Recycling data for better instruction-tuning. In NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following.

Varvara Logacheva, Daryna Dementieva, Sergey Ustyantsev, Daniil Moskovskiy, and et al. 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.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

Abdurahman Maarouf, Nicolas Pröllochs, and Stefan Feuerriegel. 2024. The virality of hate speech on social media. Proc. ACM Hum.-Comput. Interact., 8(CSCW1).

Daniil Moskovskiy, Sergey Pletenev, and Alexander Panchenko. 2024. LLMs to replace crowdsourcing for parallel data creation? the case of text detoxification. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages 14361–14373, Miami, Florida, USA. Association for Computational Linguistics.

Cicero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. 2018. Fighting offensive language on social media with unsupervised text style transfer. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 189–194, Melbourne, Australia. Association for Computational Linguistics.

Skywork o1 Team. 2024. Skywork-o1 open series. https://huggingface.co/Skywork.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, and Alec Helyar et al. 2024. Openai o1 system card. Preprint, arXiv:2412.16720.

OpenSource-O1. 2024. Open o1: A model matching proprietary power with open-source innovation. Accessed: 2025-02-01.

Mohammad Mahdi Abdollah Pour, Parsa Farinneya, Manasa Bharadwaj, and et al. 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.

Emmanuelle-Anna Dietz Saldanha and Antonis C. Kakas. 2020. Cognitive argumentation and the suppression task. CoRR, abs/2002.10149.

Anirudh Som, Karan Sikka, Helen Gent, Ajay Divakaran, and et al. 2024. Demonstrations are all you need: Advancing offensive content paraphrasing using in-context learning. In Findings of the Association for Computational Linguistics: ACL 2024, pages 12612–12627, Bangkok, Thailand. Association for Computational Linguistics.

Qwen Team. 2024a. Qwen2.5: A party of foundation models.

Qwen Team. 2024b. Qwq: Reflect deeply on the boundaries of the unknown.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, and et al Ichter. 2022. Chain-of-thought prompting elicits reasoning in large language models. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

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.

Chiyu Zhang, Honglong Cai, Yuezhang Li, Yuexin Wu, and et al. 2024. Distilling text style transfer with self-explanation from LLMs. In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop), pages 200–211, Mexico City, Mexico. Association for Computational Linguistics.

Yu Zhao, Huifeng Yin, Bo Zeng, Hao Wang, Tianqi Shi, Chenyang Lyu, and et al. 2024. Marco-o1: Towards open reasoning models for open-ended solutions. Preprint, arXiv:2411.14405.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, and et al. 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.

Yuxiang Zhou, Jiazheng Li, Yanzheng Xiang, Hanqi Yan, Lin Gui, and Yulan He. 2024. The mystery of in-context learning: A comprehensive survey on interpretation and analysis. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 14365–14378, Miami, Florida, USA. Association for Computational Linguistics.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. JudgeLM: Fine-tuned large language models are scalable judges. In The Thirteenth International Conference on Learning Representations.

A Methodology

In this section, we present the aggregation method for the Shapley values (Section A.1), the different prompts used to generate the paraphrases (Section A.2), and the detailed procedure to clean the data (Section A.3).

A.1 Shapley Value Aggregation

In Section 3 we explained that five toxicity detectors are used to generate the aggregated Shapley values. To mitigate potential issues of robustness, bias, generalization, and false positives, especially in cases of implicit toxicity, we apply different aggregation methods based on each model's prediction performance. For comments where at least one model accurately predicts the class with a probability of 0.7 or higher, we aggregate the Shapley values by selecting only the tokens that are commonly identified as toxic across all models. In cases where all models incorrectly classify the comment, we aggregate the Shapley values of all models, as they at least identify toxic words that, in other contexts, could be considered negative. By considering all extreme cases, we account not only for clearly toxic words, but also for potentially toxic words that classifiers may misclassify due to performance limitations. This adds noise helps capture subtle or context-dependent toxic elements, challenging the LLMs' contextual understanding.

A.2 R^3-Detox Instructions

In the R^3-Detox framework, several prompts have been used: prompts to generate the reasoning for the ParaDetox, APPDIA, Parallel Detoxification, and Jigsaw Unintended Bias datasets, as well as the prompt used to generate the final non-toxic paraphrases. For all prompts, we use a structured chat template. Given the Self-Reflection's own instruction prompt, we introduce our task prompt as the user message, as it was experimentally observed that failing to introduce each model's system prompt caused the model to deviate from its training data distribution.

The instructions for generating the reasoning for the toxic and non-toxic sentences are given in Figures 5 and 6, respectively. As shown in these figures, the instructions for the toxic prompts are more guided to limit potential hallucinations and the disclosure of the provided information by explicitly describing the three steps in our framework. In the case of the non-toxic instruction, the model is only instructed to explain why the sentence is not toxic by examining it for potential harmful content.

The instructions for generating the final paraphrase are presented in Figure 7. The provided prompt is divided into four elements: task description, format instructions, demonstrations, and final instructions containing the sentence and relevant information for generating the paraphrase.

A.3 Data Cleaning

As described in Section 3, we use post-processing techniques to clean and eliminate unwanted elements from the generated reasoning. These unwanted elements are statements that acknowledge that a non-toxic sentence was given as a reference to generate the Reflect, Reason, and Rephrase few-shot dataset. To eliminate the unwanted reasoning elements generated by the Self-Reflection models, we use Qwen 2.5 32B and instruct it with the prompt provided in Figures 8 (for toxic sentences) and 9 (for non-toxic sentences).

B Evaluation

In this appendix we introduce the metrics used to evaluate detoxification (Section B.1), the prompts used for evaluation with JudgeLLM (Section B.2), and the guidelines and annotation software in use (Sections B.3 and B.4).

B.1 Detoxification Metrics

In this section we provide a detailed explanation of the evaluation metrics introduced in Section 4.2, which are used to assess the quality of the generated non-toxic paraphrases. Each metric evaluates a different aspect of the detoxification process, including the preservation of meaning, fluency, and the reduction of toxicity. The following is a summary of the metrics:

- **Style Transfer Accuracy (STA)**: The percentage of non-toxic outputs identified by a style transfer model (Logacheva et al., 2022).

- **BERTScore**: We use the SimCSE (Gao et al., 2021) RoBERTa model to assess how well the model preserves the semantic meaning across tokens.

- **Content Preservation (SIM)**: The cosine similarity between the embeddings of the original toxic sentence and its paraphrase, computed using the model from Wieting et al. (2019).

- **Fluency (FL)**: The percentage of fluent sentences identified by a classifier trained on linguistic acceptability (Warstadt et al., 2019).

- **Joint Score (J)**: Quantifies the overall detoxification of the text as the product of the STA, SIM, and FL scores.

- **Toxicity Score (Tox)**: The probability that the text is toxic, as provided by the implementation of Hanu and Unitary team (2020), which was used in Som et al. (2024) to compute the toxicity reduction.

B.2 JudgeLLM Evaluation System

During our research, JudgeLLM is used to generate evaluations for both the generated reasoning and paraphrases. Due to its flexibility in handling multiple pieces of content simultaneously, we employed the following prompts: one for the “A vs. B” evaluation framework (Figures 5 and 6) and another for the triplet tournament elimination framework (Figure 10).

B.3 Manual Annotation Reasoning

For the manual annotation of the generated reasonings, we use the Argilla Annotation framework deployed on the Hugging Face Hub due to its ease of integration and flexibility. In Figure 11, we present the interface of the annotation software. As mentioned in Section 4, we implement a “A vs B” tournament in which annotators select the best reasoning based on the following guidelines:

In this dataset, you will be presented with a prompt and two reasoning responses. Your task is to select the reasoning response that best fits the given prompt. If both reasoning responses are equally good or bad, select 0. 1 corresponds to Model A and 2 corresponds to Model B. Please rank the reasoning response based on the following criteria:

1. *Task Completion: Does the reasoning correctly tackle the task given in the prompt? For prompts involving toxic sentence analysis, does the reasoning tackle the following goals: (a) analyze words that may convey toxic meaning, (b) provide a verdict on whether it is toxic, and (c) if toxic, reason correctly on how to obtain the final paraphrase.*

2. *Coherence: Is the reasoning response coherent and logically sound? Does it contain contradictions or broken logical flow or misleading statements?*

3. *Fluency: Is the reasoning response fluent and easy to understand? Does it contain any grammatical errors?*

4. *Consistency: In the cases where a toxic sentence is analyzed, does the reasoning stay consistent with the provided paraphrase?*

Given a toxic sentence and its non-toxic paraphrase, your task is to create a reasoning process for generating the paraphrase. Avoid addresing the provided toxic sentence as "toxic sentence", just call it the "provided sentence".

1. Examine the words or phrases in the provided sentence that convey toxic behavior, and explain why they are toxic within the given context. To support this analysis, words that might indicate toxicity may be provided, but avoid explicitly mentioning that these words were given.
2. Provide a concise explanation of why the sentence exhibits toxic behavior based on your analysis.
3. Based on the toxic word analysis and the modifications present in the paraphrase, analyze the changes required to transform the sentence into the non-toxic paraphrase. Begin by identifying the changes without directly stating that the non-toxic paraphrase was provided, and later explain how these changes remove the toxicity.

Be concise in your response. Here is the information to analyse:
Toxic sentence: {sentence}
Relevant words: {toxic_words}
Label: {label}
Paraphrase: {paraphrase}

Figure 5: Instruction used to extract the Reflect, Reason and Rephrase from the APPIA, ParaDetox and Parallel Detoxification datasets.

Provide a brief explanation of why the given sentence is not toxic by first analyzing it for any harmful, offensive, or inappropriate language.To support your reasoning, potential toxic words may be provided if present. Do not reference external information, prior analysis, or the fact that words were provided.
Sentence: {sentence}
Possible toxic words: {toxic_words}

Figure 6: Instruction used to extract the Reflect, Reason and Rephrase from the non-toxic dataset.

5. *Plausibility: Is the reasoning response plausible and realistic? Is the contextual analysis provided by the models believable given the task?*

6. *Creativity: Does the reasoning response show creativity and originality?*

B.4 Manual Annotation Paraphrase

Similarly to the reasoning process, the manual annotation of the generated paraphrases is done by resorting to the Argilla Annotation framework in the Hugging Face Hub. Figure 12 presents the interface of the annotation software. As mentioned in Section 4, we employ a triplet elimination tournament in which annotators rank the best paraphrase based on the following guidelines:

You will be presented with a prompt and three model responses. Your task is to select the non-toxic paraphrase that best fits the criteria. Only one response can be ranked as the best. Rank the paraphrases based on the following criteria, where the best value is a 5 and the worst is a 1:

5. A perfect meaning-preserving, inoffensive paraphrase that not only preserves the meaning but also makes the least modifications to the original sentence.

4. A perfect meaning-preserving, inoffensive paraphrase but with significant modifications to the original sentence.

3. A paraphrase that is inoffensive but somewhat distinct in meaning from the original.

2. A meaning-preserving paraphrase that is somewhat similar in offensiveness to the original.

1. A paraphrase that is very different in meaning and not less offensive than the original.

C Results

In this section, we present examples and additional metrics obtained for the research questions.

C.1 RQ2: Can Self-Reflection models Reflect, Reason and Rephrase?

In Table 3 and 4, we present reasoning examples of our R^3-Detox framework for toxic and non-toxic comments, part of our few shot examples dataset generated in Section 6.2.

C.2 RQ3: Reflect, Reason, Rephrase ICL vs State-of-Art

In this section, we examine the metrics that measure the influence of the demonstrations on the different metrics discussed in Section C.2.1, as well as the hard examples observed during the annotation process in Section C.2.2.

````
Provided a toxic sentence and words that may convey toxic meaning, your task is to generate a non-toxic rephrased version of the toxic sentence while maintaining the same meaning and matching the target style/attribute as much as possible. During your task, generate step-by-step reasoning following these instructions:

1. Examine the words or phrases in the provided sentence that convey toxic behavior, and explain why they are toxic within the given context. To support this analysis, words that might indicate toxicity may be provided.
2. Provide a concise explanation of why the sentence exhibits toxic behavior based on your analysis.
3. Based on the toxic word analysis, analyze the changes required to transform the sentence into a non-toxic paraphrase. Begin by identifying the changes needed and later explain how these changes remove the toxicity.


The final output must be the following plain text:
 ```Final Reasoning: The reasoning process generated
Final Paraphrase: The final paraphrase generated```

**Example {str(idx+1)}:**
Toxic sentence: {sentence}
Possible toxic words: {toxic_words}
```Final Reasoning:"{reasoning}"
Final Paraphrase:"{paraphrase}"```

Generate a non-toxic rephrased version of the toxic sentence while maintaining its meaning and matching the target style or attribute as closely as possible. Provide your final response/output in English, strictly adhering to the specified JSON format.

Toxic sentence: {sentence}
Possible toxic words: {toxic_words}
````

Figure 7: Instruction used to detoxify the given toxic sentence.

C.2.1 Importance of Number of Demonstrations

In Figures 13 to 19, we present the metrics discussed in Section 4.2. We observe that as the number of demonstrations increases, the text similarity improves both semantically and lexically, while also reducing its toxicity score. The only metric that does not improve with demonstrations is the Style Transfer Accuracy (STA), due to the rudeness remaining in some sentences.

C.2.2 Correlation Between Metric Rankings and Annotator Evaluations

In Section 6.3.3, we discuss how the annotation process was highly subjective, with annotators ranking sentences differently. Table 5 presents the comments that exhibited the most disagreement during the annotation process. From this table, we can observe that the annotation disagreements primarily originated from subtle differences between the evaluated paraphrases or from not knowing the specific meaning of words such as "sandngr." A clear example of a paraphrase where an annotator needs a proper understanding to pick up on subtle differences between incorporating one word or another is in the case of "that's just a bitch move," where paraphrases change it to "bad," "low," or "cheap move." To properly select the correct one, annotators should have a deep understanding of the English language to choose the most appropriate word. Another interesting example is "It would be fucking illegal as shit to straight-up kill someone," where the models, in some cases, change "kill" to "harm," which makes it less aggressive, or change "fucking" to "very" or "absolutely," which also reduces the tone of the phrase. These small modifications can affect the selection of the most suitable paraphrase, leaving the decision to the subjectivity of the annotator. With these examples, we emphasize the subjectivity and deep contextual understanding needed to correctly annotate the detoxification paraphrases of highly capable models.

Additionally, Figure 20 shows the p-value matrix corresponding to the Spearman's rank correlation

System prompt:
Your task is to extract and remove information from a reasoning process. The information to be remove is the following:

- Omit any part where the model explicitly states the task it is performing or the fact that it need to analyse the toxic sentence and the provided non-toxic sentence.
- Remove the sentences where the model acknowledges that the sentence is toxic before doing the toxic words analysis. Instead change toxic sentence for the provided sentence.
- In the part where it is explained how to make the changes in the sentence to generate a non-toxic paraphrase, rewrite it to present the information as though the non-toxic paraphrase was not given to the model, with the steps described as part of its inherent process.
- When discussing the changes that will need to be made to the original toxic talk, use future tense.
- Correct any words with random capitalization.\
- Rewrite the sentences to remove the mention that some toxic words were provided and instead state that the potential toxic words are identified, or in cases where they mention examining the identified toxic words, replace it with 'the words that struck as toxic are the following'.
- Translate any chinese characters to english.
- Once you have removed the unnecessary information, you generate the original reasoning process without that information in the corresponding JSON format: ```json\n{"reasoning": "extracted reasoning"}\n```

User prompt:
Reasoning Process:
"{reasoning}"

Figure 8: Instruction to eliminate unwanted sentences from the generated Reflect, Reason and Rephrase reasonings for the toxic sentences.

System prompt:
Your task is to extract and remove information from a reasoning process. The information to be remove is the following:

1. Omit any part where the model explicitly states the task it is performing.
2. Eliminate any comments that stated that first it must define what is toxicity.
3. Rewrite any sentence that states that 'I need to determine why the provided sentence is not toxic' with 'I need to determine the toxicity of the sentence'.
4. Rewrite the sentences to remove the mention that some toxic words were provided and instead state that the potential toxic words are identified, or in cases where they mention examining the identified toxic words, replace it with 'the words that struck as toxic are the following'
5. Translate any chinese characters to english.

Once you have removed the unnecessary information, you generate the original reasoning process without that information in the corresponding JSON format: ```json\n{"reasoning": "extracted reasoning"}\n```.
User prompt:
Reasoning Process:
"{reasoning}"

Figure 9: Instruction to eliminate unwanted sentences from the generated Reflect, Reason and Rephrase reasonings for the non-toxic sentences.

Given a toxic sentence and words that may convey a toxic meaning, your task is to generate a non-toxic rephrased version of the sentence while maintaining the same meaning and matching the target style or attribute as closely as possible. A perfect non-toxic paraphrase is one that perfectly preserves the meaning, is inoffensive, and makes the least modifications to the original sentence. Avoid generating a paraphrase that either preserves the meaning but retains some offensiveness or is inoffensive but significantly alters the original meaning.
Toxic sentence: {sentence}
Possible toxic words: {toxic_words}

Figure 10: Prompt used to evaluate the generated paraphrases by JudgeLLM.

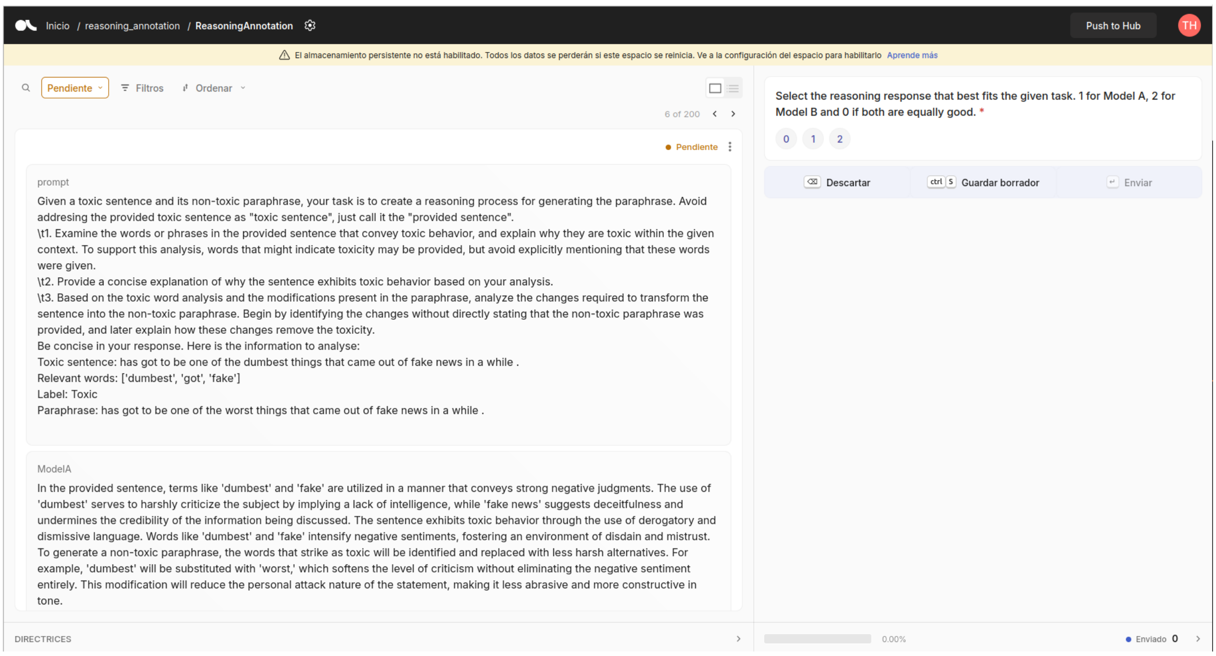

Figure 11: Argilla annotation framework for reasoning evaluation.

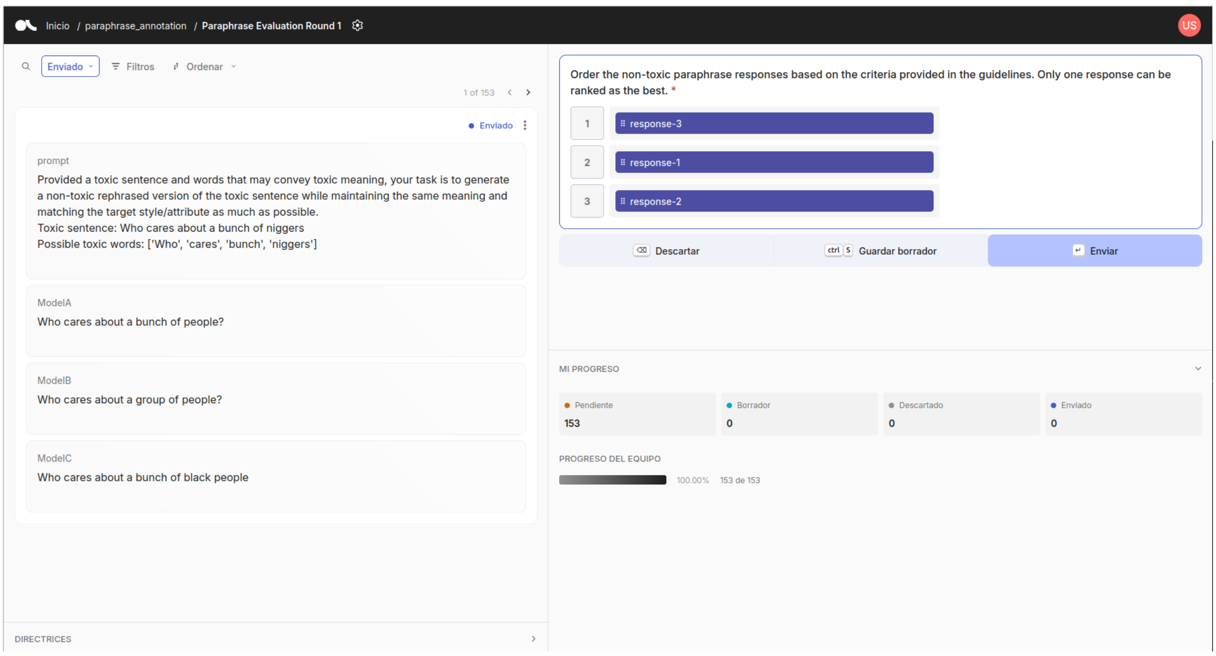

Figure 12: Argilla annotation framework for paraphrase evaluation.

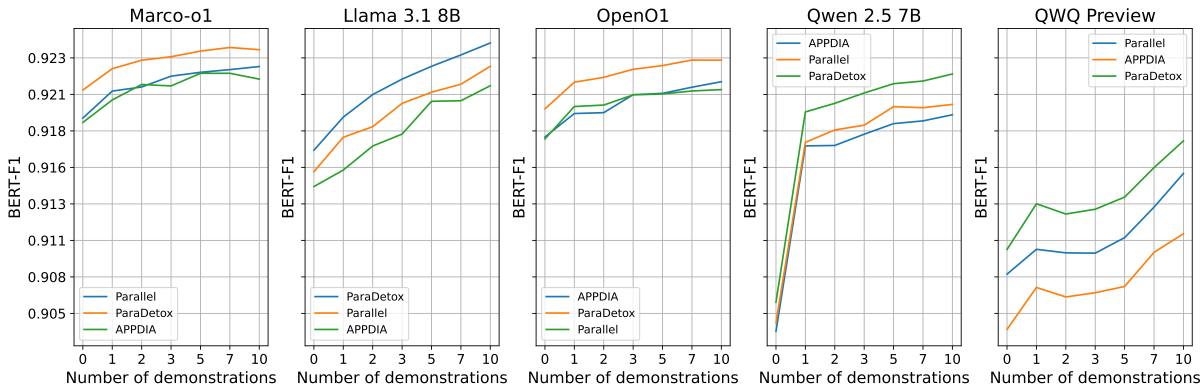

Figure 13: BERT F1 as a function of the number of demos. The performance increases when demos are introduced.

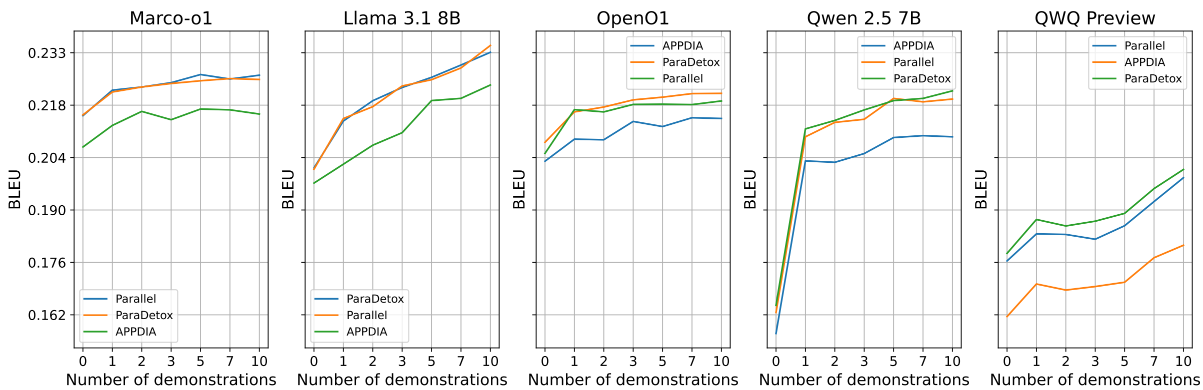

Figure 14: BLEU as a function of the number of demos. The performance increases when demos are introduced.

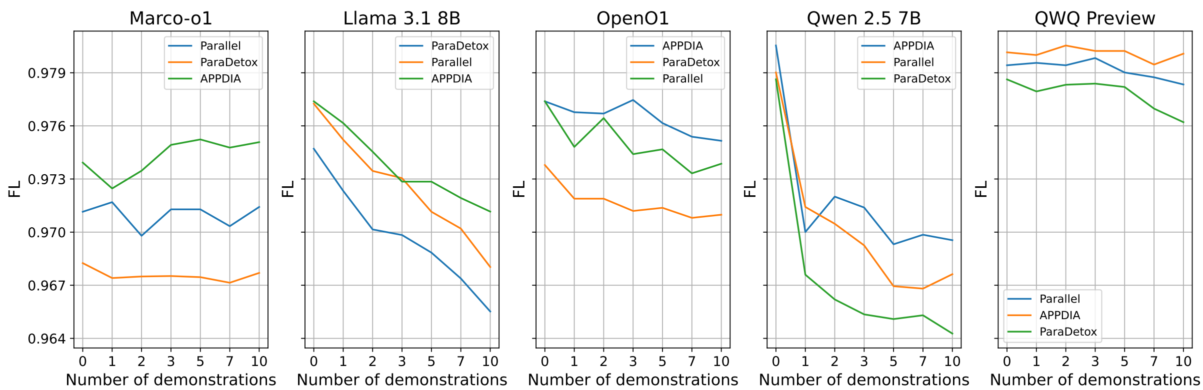

Figure 15: Fluency as a function of the number of demos. The performance decreases when demos are introduced, as the sentences become more similar to the one provided.

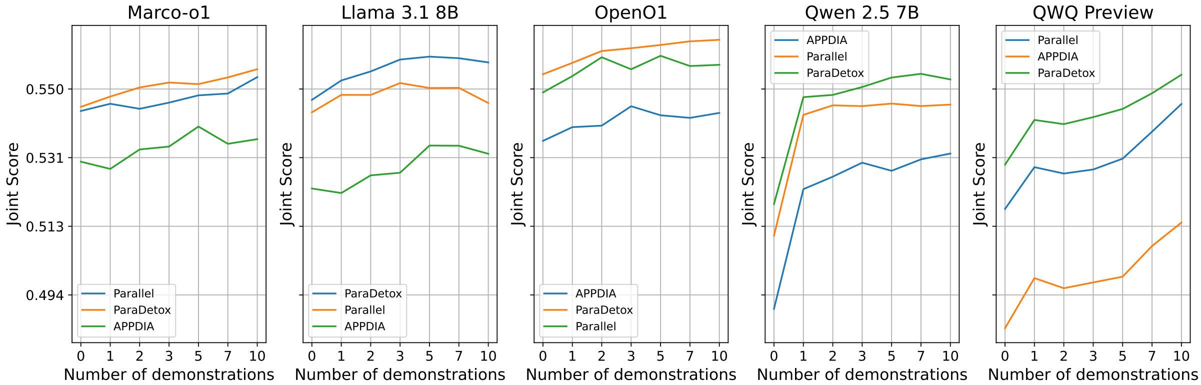

Figure 16: Joint Score as a function of the number of demos. The performance increases when demos are introduced, but the rate of increase depends on the model used.

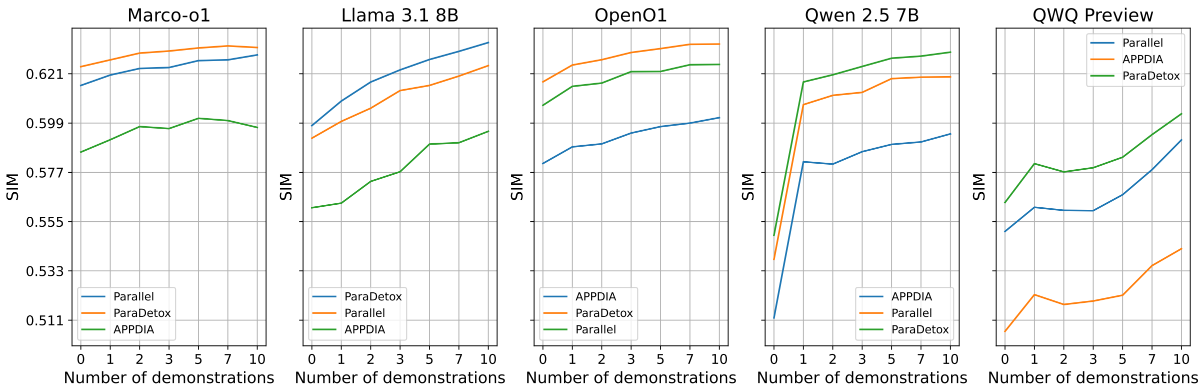

Figure 17: Content Similarity as a function of the number of demos. The performance increases when demos are introduced.

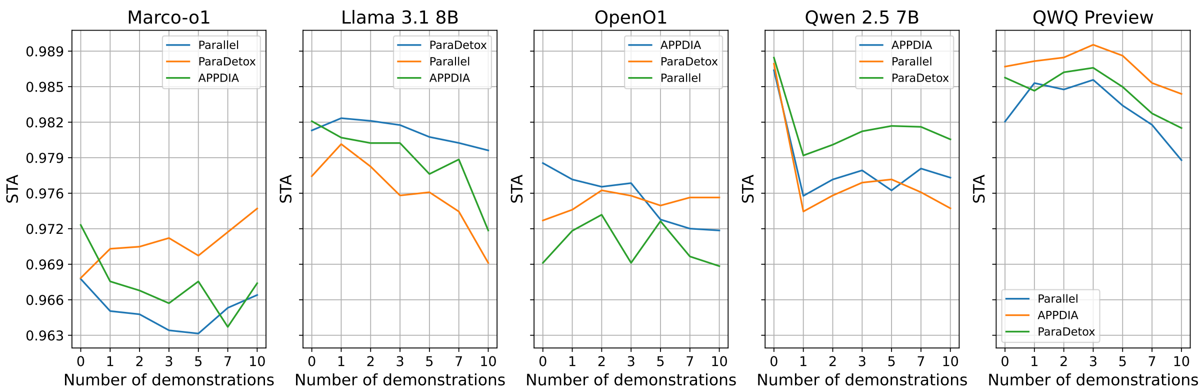

Figure 18: Style Transfer Accuracy as a function of the number of demos. Performance decreases when demos are introduced, as some text still contains instances of rudeness.

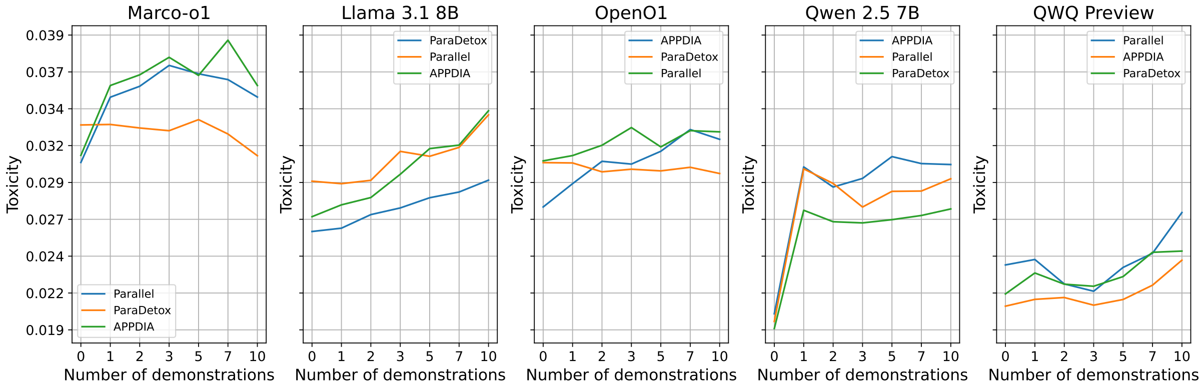

Figure 19: Toxicity Score as a function of the number of demos. Performance increases when demos are introduced.

matrix presented in Section 6.3.3. From this figure, we can observe that the p-values are greater than 0.05, indicating a lack of significant agreement between the annotators.

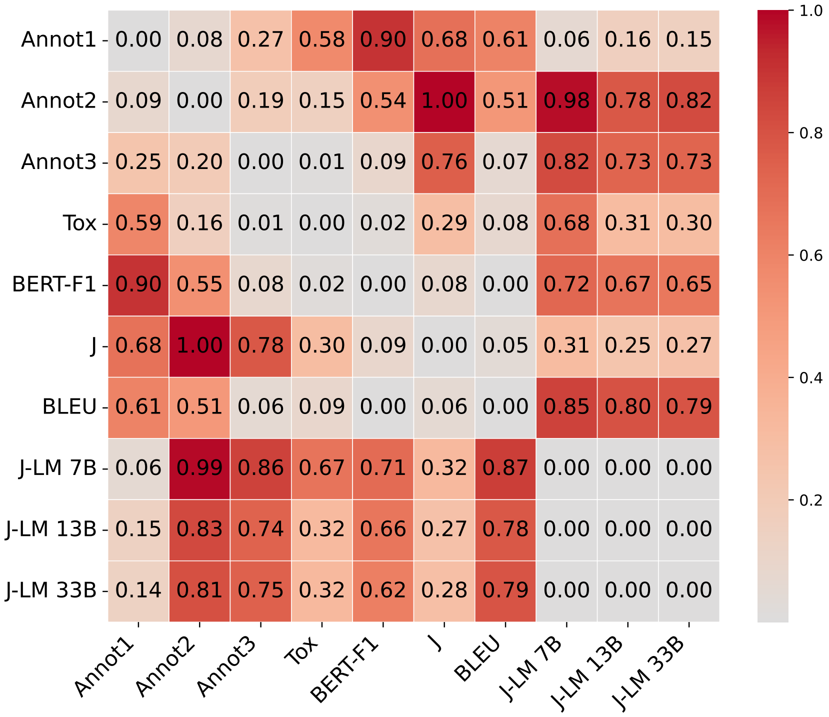

Figure 20: P-value matrix for the Spearman's rank correlation between the metrics to evaluate the detoxification quality.

Comment	Label	Reasoning
i know where this ass lives, he is a complete asshat and acts like an entitled bitch.	Toxic	The sentence 'I know where this ass lives, he is a complete asshat and acts like an entitled bitch.' contains several words that strike as toxic: 'ass,' 'asshat,' and 'bitch.' These terms are often used in a derogatory way, especially when referring to someone's personality or behavior. 'Ass' is used to refer to a person in a disrespectful and objectifying manner. 'Asshat' is an insult used to describe someone as rude or unpleasant, which dismisses their opinions or actions without addressing the issues. 'Bitch' is used to criticize someone's behavior, adding to the toxic nature of the sentence. The word 'complete' emphasizes the extent to which the person is seen as an asshat, which is an oversimplification and not respectful. 'Entitled' describes someone who believes they are deserving of special privileges without earning them, and when coupled with derogatory terms, it makes the sentence toxic. The sentence is toxic because it uses vulgar, insulting language to demean and dismiss someone's behavior without any attempt at understanding or constructive criticism. It's all about tearing the person down rather than addressing specific issues. To generate a non-toxic paraphrase, we will replace the vulgar reference with a simple pronoun, such as changing 'this ass' to 'he.' We will also transform 'is a complete asshat' into 'doesn't behave well,' which is a more factual and less emotionally charged way of describing the issue. Additionally, we will omit 'acts like an entitled bitch' entirely and end the sentence with 'he doesn't behave well,' which is a general statement without the specific derogatory comparisons. The changes will involve removing the vulgar and insulting language and replacing them with more neutral, factual statements about the person's behavior. This approach will be less likely to offend or upset others and will focus on the behavior rather than attacking the person's character. The paraphrase 'I know where he lives; he doesn't behave well.' maintains the core message about the person's behavior but does so in a much more respectful and less offensive manner.
Someone please tell Mulcair to butt out. He's not being helpful.	No Toxic	The sentence is: 'Someone please tell Mulcair to butt out. He's not being helpful.' I need to determine the toxicity of the sentence. Toxic language is usually harmful, offensive, or inappropriate. It can involve insults, derogatory terms, or language that demeans someone based on their characteristics or actions. The words that struck as toxic are the following: 'butt', 'tell', and 'not'. 'Butt out' is an idiomatic expression meaning to stop interfering or mind one's own business. It's informal and somewhat abrupt, but it's not necessarily offensive unless directed in a malicious way. 'Tell' is a neutral verb meaning to communicate information to someone. In this sentence, 'someone please tell Mulcair,' it's a request for someone else to convey a message to Mulcair. There's nothing inherently toxic about telling someone something. 'Not' is a negation used to express that something is not happening or being done. Here, it's used in 'He's not being helpful,' which is a statement expressing that Mulcair's actions are not contributing positively to a situation. Overall, the sentence expresses disappointment or frustration with Mulcair's involvement, suggesting that he should stop interfering because his input is not helpful. While it's direct and perhaps a bit rude, it doesn't cross into toxic language unless the context suggests otherwise. Toxic language would involve personal attacks, insults, or language that demeans based on characteristics like race, gender, etc. In this sentence, there's no personal attack beyond expressing dissatisfaction with his actions. Therefore, this sentence isn't toxic; it's just a strong expression of discontent with someone's behavior.

Table 3: Example of Toxicity Analysis and Paraphrasing

Comment	Label	Reasoning
the chicken is dryer than an 76 year old pussy .	Toxic	The sentence ’The chicken is dryer than an 76 year old pussy.’ contains inappropriate and offensive language. The words that struck as toxic are ’pussy’ and ’old’. ’Pussy’ is likely used in a derogatory manner, and ’old’ combined with ’pussy’ is disrespectful. The sentence is toxic because it uses derogatory terms to make a comparison, which is disrespectful and potentially ageist and sexist. It ties a negative attribute (being dry) to an older person, reinforcing negative stereotypes about aging, and uses sexual terminology in a disparaging way, which is generally considered toxic because it objectifies and dehumanizes the subject. To generate a non-toxic paraphrase, we will need to eliminate the offensive language and comparison, simplify the sentence to directly describe the chicken, and correct the grammar (assuming ’76’ was a typo or mistake). The paraphrase ’The chicken is very dry.’ removes any toxicity by avoiding disrespectful or derogatory terms and focusing solely on the description of the chicken. It is clearer, more respectful, and does not perpetuate any negative stereotypes.
that should say ’birthed by white men’.	No Toxic	I need to determine the toxicity of the sentence: ’that should say ’birthed by white men’.’ The words that struck as toxic are the following: [’birthed’]. In standard English, ’birthed’ is a neutral term meaning to give birth and doesn’t inherently carry any harmful or offensive connotations. However, the context in which it’s used can influence its perception. In this sentence, ’birthed’ is used in the phrase ’birthed by white men,’ which suggests a specific group of people as the subject of birth. Considering the term ’white men,’ this phrase can be problematic because it implies a particular racial or gendered perspective. The use of ’white men’ as a descriptor can be seen as exclusionary or biased, potentially marginalizing individuals who do not identify as white or male. This could be interpreted as toxic if it’s part of a larger narrative that promotes discrimination or exclusion. However, the sentence itself doesn’t explicitly contain any harmful language. It’s a statement that should be replaced with ’birthed by white men.’ The issue arises from the choice of words and the context in which they’re used, rather than the sentence being toxic in and of itself. The potential toxicity lies in the implication of the phrase ’birthed by white men,’ which can be seen as biased or exclusionary. To further analyze, if the intention behind the sentence is to highlight a specific group’s role in a particular context, it’s essential to ensure that the language used is inclusive and respectful. Using terms like ’birthed by white men’ without proper context can lead to misunderstandings or reinforce stereotypes. In conclusion, while the sentence doesn’t contain overtly toxic language, the choice of words and the context in which they’re used can contribute to a toxic environment if not handled carefully. It’s crucial to consider the broader implications and ensure that language is used in a way that promotes inclusivity and respect.

Table 4: Example of Toxicity Analysis and Paraphrasing

Original Sentence	Paraphrase 1	Paraphrase 2	Paraphrase 3
that 's just a bitch move.	that 's just a bad move.	That's just a cheap move.	that 's just a low move.
You're fucking insane	You're being unreasonable.	You're extremely unreasonable.	You're insane.
Lol check the context dipshit stop profiting from child labor	Check the context. Let's discuss the issue of child labor seriously.	Hey, it would be great if you could check the context of this situation. Let's avoid using derogatory language and instead focus on finding a solution that doesn't involve exploiting children.	Lol check the context. Stop exploiting children through labor.
Even more important, they're a sovereign white (Slavic) country, not some sandngr shit hole.	Even more important, they're a sovereign white (Slavic) country, not a place with certain challenges.	Moreover, it's a sovereign Slavic country, not like some chaotic desert regions.	Even more significantly, they are an independent white (Slavic) nation, not a country with a troubled history.
It would be fucking illegal as shit to straight up kill someone.	It would be absolutely illegal to harm someone straight up.	It would be illegal to straight up kill someone.	It would be very illegal to straight up kill someone.
It would be fucking illegal as shit to straight up kill someone.	It would be illegal to harm someone directly.	It would be illegal to directly kill someone.	It would be illegal to kill someone.

Table 5: Annotation examples with high disagreement.