Improving Covert Toxicity Detection by Retrieving and Generating References

Dong-Ho Lee[1,2], Hyundong Cho[2], Woojeong Jin[2], Jihyung Moon[1], Sungjoon Park[1], Paul Röttger[3], Jay Pujara[2], Roy Ka-Wei Lee[4]
[1]SoftlyAI Research, [2]University of Southern California,
[3]Bocconi University, [4]Singapore University of Technology and Design
{dongho.lee,woojeong.jin}@usc.edu {jcho,jpujara}@isi.edu
{jihyung.moon,sungjoon.park}@softly.ai {paul.rottger}@unibocconi.it {roy_lee}@sutd.edu.sg

Abstract

Models for detecting toxic content play an important role in keeping people safe online. There has been much progress in detecting overt toxicity. Covert toxicity, however, remains a challenge because its detection requires an understanding of implicit meaning and subtle connotations. In this paper, we explore the potential of leveraging *references*, such as external knowledge and textual interpretations, to enhance the detection of covert toxicity. We run experiments on two covert toxicity datasets with two types of *references*: 1) information retrieved from a search API, and 2) interpretations generated by large language models. We find that both types of references improve detection, with the latter being more useful than the former. We also find that generating interpretations grounded on properties of covert toxicity, such as humor and irony, lead to the largest improvements[1].

1 Introduction

The proliferation of toxic speech on social media platforms has raised significant societal concerns. Previous attempts to detect such content have largely focused on *overt expressions* (Waseem and Hovy, 2016; Davidson et al., 2017; Founta et al., 2018; Basile et al., 2019), and often rely on apparent associations, such as explicit language, overlooking contextual nuances (Röttger et al., 2021; Hartvigsen et al., 2022; Lee et al., 2022). In reality, however, toxicity is often more latent than apparent. This underscores the importance of identifying these concealed forms of toxicity, i.e. *covert toxicity*, which includes implicit expressions that convey prejudiced views towards specific groups (Breitfeller et al., 2019; Han and Tsvetkov, 2020) and masked forms that utilize coded language and emojis (Taylor et al., 2017; Lees et al., 2021). Therefore, detecting covert toxicity requires deciphering connotations and contextual cues, posing a significant challenge to existing toxicity classifiers (Ocampo et al., 2023).

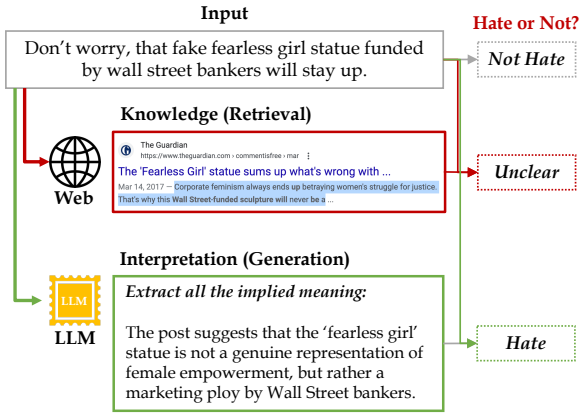

Figure 1: Covertly toxic statements are not immediately apparent and may be challenging for existing toxicity classifiers. Relevant references, such as retrieved documents or generated interpretations, can aid detection.

Recent studies have demonstrated that complex and multi-layered tasks, such as fact checking and question answering, can be enhanced by an intermediary stage of relevant document retrieval (Karpukhin et al., 2020; Lewis et al., 2020; Izacard and Grave, 2021; Singh et al., 2021; Liu et al., 2023; Gao et al., 2023; Li et al., 2023) or generating reasoning steps (Zhou et al., 2022; Wei et al., 2022; Kojima et al., 2022; Wang et al., 2023a). We focus on identifying covert toxicity, and, in a similar vein, we propose that augmenting models with an intermediate step of identifying *references* would enhance their performance in detecting covert toxicity. To illustrate, consider the example in Figure 1, where the input text ("*Don't worry, that fake fearless girl statue funded by wall street bankers will stay up*") is not overtly toxic, which makes it challenging to detect. However, we can provide additional contextual cues by utilizing two types of *references*: (1) **Web-retrieved external knowledge** can provide contextual cues linking

[1]https://github.com/softly-ai/RefBasedToxicityDetector

Proceedings of the The 8th Workshop on Online Abuse and Harms (WOAH), pages 266–274

the "*fearless girl statue*" to feminism, albeit not overtly. The model could recognize the association between the statue and feminism, yet results from `gpt-3.5-turbo` remain inconclusive, indicating ambiguity. (2) **Large language model (LLM) - generated interpretation** can reveal underlying connotations when prompted (*"Extract all the implied meaning behind the text."*). By integrating such interpretations into the model, it can better comprehend the contextual implications embedded within a text, thereby facilitating a more accurate prediction.

In this work, we explore the efficacy of references for covert toxicity detection and examine the capability of LLMs to *generate* references that are as effective as the documents they can generate for tasks demanding comprehensive knowledge (Yu et al., 2023). We compare search results from the web with interpretations obtained from simple prompts for LLMs to uncover hidden meanings in the given text, in terms of their ability to aid toxicity detection. We show that interpretations generated from our pipeline with LLMs are the most effective, and that the effectiveness of these interpretations can be further improved by grounding their prompts to ask about specific properties of covert toxicity (Ocampo et al., 2023).

In summary, we show that (1) web-retrieved external knowledge and LLM-generated interpretations help models make more accurate predictions on covert toxicity; (2) LLM-generated interpretations related to granular properties of covert toxicity are the most effective references.

2 Core Concepts

2.1 Covert Toxicity

Covert toxicity encompasses various forms of hidden toxicity that may not be immediately apparent (Lees et al., 2021). It includes *implicit* and *subtle* toxic speech, which does not overtly express abusive or hateful intent. Instead, it relies on unique nuances that mask the true meaning beneath the surface (ElSherief et al., 2021). Covert toxicity conveys messages that are delicate or elusive, making them challenging to analyze or describe. It often relies on indirect methods like complex sentence structures or emojis to convey its meaning (Ocampo et al., 2023).

Detecting covert toxicity presents two main challenges. The first is understanding hidden toxicity in language that deliberately avoids explicit profanity and insults. In such cases, people may attempt to conceal their toxicity through obfuscation tactics such as misspellings, code words, implied references, or utilize visual signs such as emojis and ASCII art) or subtle harmful expressions like irony, sarcasm, and microaggressions. To improve detection in these cases, it is crucial to comprehend the underlying meaning behind the words used. The second is the risk of misclassifying positive statements as toxic due to spurious correlations, such as identity-specific terms, without considering the context. To avoid such errors, the detector needs to adeptly understand the contextual cues surrounding specific terms.

2.2 References

This paper proposes to employing helpful *references* to improve covert toxicity detection. We propose two distinct types of references with regard to the input text q. (1) **Non-parametric references** refer to web-retrieved external knowledge that can be obtained from an external corpus or the web relating to q. Retrieval of this information typically involves identifying the most semantically similar document $\mathcal{D}$ to q; (2) **Parametric references** refer to LLM-generated interpretation that can be generated from instruction-following LLM $\mathcal{M}$. Given query q, $\mathcal{M}$ is prompted to produce an intermediate output, denoted as $\mathcal{G}_i \sim \mathcal{P}_{\mathcal{M}}(\mathcal{G}_i \mid i, q)$, where i is a specific instruction. Based on different i, intermediate output $\mathcal{G}_i$ can contain different information. We use the properties that are frequently observed in covert toxicity according to (Ocampo et al., 2023), such as black humor, irony, and rhetorical questions, and experiment with various combinations of the generated references. We share the specific wording for each prompts in Appendix Table 5.

3 Experiments

3.1 Datasets

In order to demonstrate the efficacy of our framework in detecting different forms of covert toxicity, we evaluate on two distinct covert toxicity detection datasets. (1) **Latent Hatred (ElSherief et al., 2021)** is a binary classification task that involves identifying whether a given text contains implicit hate; (2) **Hatemoji (Kirk et al., 2022)** is a binary classification tasks that involves determining whether the short-form synthesized statement contains emoji-based hate speech. Dataset details are discussed in Appendix A.1.

Prompt Strategy	Latent Hatred	Hatemoji
	Binary F1	Binary F1
Direct	0.593	0.873
Chain-of-Thought	0.572	0.845
Reference	**0.615**	**0.875**

Table 1: **LLM-based zero-shot performance** comparison. The best model for each dataset is shown in **bold**. 'Implication' property of reference has been used.

Model	Input	Latent Hatred
		Binary F1
BERT-base	Text	0.683
RoBERTa-large	Text	0.733
BERT-base	Text + Reference	**0.709**
RoBERTa-large	Text + Reference	**0.742**

Table 2: **Supervised training performance** comparison. The best model for Latent Hatred is shown in **bold**. 'Implication' property of reference has been used.

3.2 Baselines & Implementation Details

Zero-shot Evaluation using LLMs. In our evaluation, we contrast our methodology with the following techniques: (1) **Direct** simply requests the prompt to produce the outcome; (2) **CoT** uses chain-of-thought prompts (Kojima et al., 2022; Wei et al., 2022) to generate both an explanation and its corresponding response; (3) **Reference** is our main approach that leverages *references* to produce the outcome. We have five different properties of *reference* which are implication, sentiment, irony, humor and rhetorical question. The reference property used for Table 1 and Table 2 is implication, where all implied meanings of the target are generated. We share our prompts and implementation details in Appendix A.2.1.

Supervised Training. We present two baselines for supervised training: (1) **Text** learns the direct mapping between the target text and its corresponding label; while (2) **Text + Reference** trains a model to map the concatenation of target text and its corresponding reference to its respective label. Implementation details are in Appendix A.2.2.

4 Experimental Results

4.1 Performance Comparison

Zero-shot Inference. Table 1 indicates that the reference-based approach is highly effective in improving the zero-shot performance of LLM. On the other hand, the use of a chain-of-thought style approach for tasks with implied meaning is found to be counterproductive, as it leads to a decrease in performance. This finding is in contrast to the effectiveness of this approach for tasks that require complex reasoning, such as math or logical reasoning tasks (Wei et al., 2022). Notably, the performance difference between the reference-based and non-reference-based approaches is significant for implicit toxicity, while it is relatively small for Hatemoji, where the input text mostly consists of explicit toxic content, although it may be hidden within emojis. It is important to highlight that the p-value is approximately .000, indicating a significant result (See Appendix A.2.1 for more details).

Supervised Training. The results presented in Table 2 demonstrate that the model trained on both the target text and the *reference* exhibits superior performance compared to those trained solely on the target text, with a notable 1.2 - 3.6% increase in binary F1. The evidence suggests that incorporating supplementary information into the fine-tuning process leads to an enhancement in performance.

4.2 Impact of Reference Type

In order to comprehensively evaluate the impact of reference types on performance, we compare the set of references described in Section 2.

Non-parametric vs. Parametric References. To start, we compare the use of non-parametric and parametric references. For the non-parametric reference, we initiate a request to the Google Search API using the input text q directly as a search query. We gather the top five search results and concatenate their descriptions to generate a passage via LangChain (Chase, 2022). The resulting passage is then utilized as a *reference*. For the parametric reference, we use implication which is used in Table 1 and 2. Figure 2 indicates a noticeable improvement in performance when using both parametric and non-parametric references. However, it is worth noting that the use of parametric reference outperforms non-parametric reference by a significant margin of 2.1%.

Variations of Parametric References. To further investigate what other parametric references can be generated to help model prediction, we employ few properties (*i.e.,* implication, sentiment, irony, humor, rhetorical question) of implicit hate speech (Ocampo et al., 2023). Prompts for generating reference for each property are in Table 6. Figure 2 shows the varying effectiveness of the

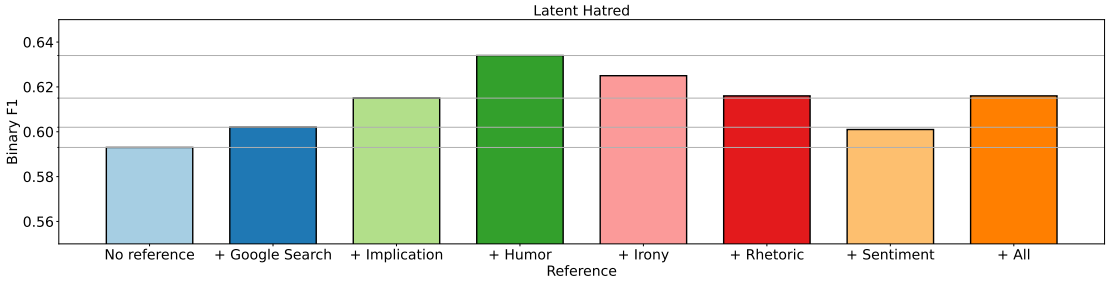

Figure 2: **LLM-based zero-shot performance** comparison with different reference variants on Latent Hatred. `+all` refers to the concatenation of all the references (*i.e.,* implication, humor, irony, rhetoric, sentiment).

type of generated references we use. Results indicate that interpretations with prompts that ask about granular properties of covert toxicity (*e.g.,* humor, irony) are the most effective references. We could not reveal any specific performance improvement patterns, but one interesting finding pertains to the sentiment reference. Sentiment is usually expressed as positive or negative while there is no strong positive correlation between negative sentiment and implicit hate, which may contribute to the poor performance observed in this aspect.

4.3 Generated Interpretations vs Human-written Implications

We proxy the quality of our generated interpretations by comparing them with the human-written implications in the Latent Hatred dataset. Since the human-annotated implications are only provided for a subset of those that are labeled as containing implicit hate, we compute accuracy only for these samples in the zero-shot setting. For the model interpretations, we use 'implication' property of the *reference*. On the surface, Table 3 indicates that human implications are better predictors of convert toxicity than model interpretations. However, the former were written by annotators who knew the label of the instance that they were annotating, possibly introducing label leakage. Indeed, even if we only keep the human implications, accuracy remains the same. On the other hand, model interpretations are generated without knowing the label, and therefore are not biased towards generating an interpretation that hints at the ground truth. This is supported by the larger drop in accuracy when we use only model interpretations as the input.

Approach	Latent Hatred
	Accuracy
Target + Human implications	0.98
Human implications only	0.98 (−0.0)
Target + Model interpretations	0.88
Model interpretations only	0.78 (−0.10)

Table 3: **LLM-based zero-shot performance** with human implications vs model interpretations for the subset of Latent Hatred that is labeled as implicit hate.

5 Related Work

Beyond Explicit Toxicity. Focusing solely on identifying explicit harmful text content may not offer a comprehensive understanding of the nuanced intentions and societal implications associated with toxic language usage (Jurgens et al., 2019; Rossini, 2022). Recent analyses have adopted fine-grained criteria, including implication (Taylor et al., 2017; Breitfeller et al., 2019; Han and Tsvetkov, 2020; Lees et al., 2021; ElSherief et al., 2021), context sensitivity (Pavlopoulos et al., 2020; Xenos et al., 2021; Gong et al., 2021; Menini et al., 2021; Moon et al., 2023), and subjectivity (Sap et al., 2022; Rottger et al., 2022), to gain a holistic understanding of toxicity beyond explicit signs.

Enhancing Models with LLM Output. Recent research has emphasized the use of LLMs to produce contextual information, such as explanations or knowledge, for addressing specific queries. This approach involves generating intermediate reasoning stages or rationale-like explanations to tackle complex tasks (Wei et al., 2022; Kojima et al., 2022; Anil et al., 2022; Dohan et al., 2022; Wang et al., 2023b; Saparov and He, 2023). Furthermore, LLMs are employed to generate relevant knowledge for solving tasks that involve commonsense reasoning (Liu et al., 2022; Fang et al., 2022) or tasks that require knowledge (Yu et al., 2023).

6 Conclusion

In this paper, we propose a *reference*-guided covert toxicity detection framework. The framework comprises non-parametric and parametric references that can be obtained from external sources and large language models, respectively. Our study demonstrates that incorporating additional references improves the model's ability to identify covert toxicity, resulting in more accurate detection performance.

7 Limitations

The covert toxicity datasets (*e.g.,* Latent Hatred, Covert Toxicity) exhibit significant subjectivity. In a non-trivial number of cases that we manually examined, the discrepancies between LLM-based predictions and ground truth labels presented a challenge for the authors on whether the predictions or the given labels were correct. Therefore, an important future work will be to account for these cases to more accurately capture the performance of coverty toxicity detection.

(Huang et al., 2023) also mentions that individuals tend to exhibit a preference towards ChatGPT inferences in cases where there are disagreements between ChatGPT and human labels. Consequently, this may be the reason why zero-shot LLM inference demonstrates lower performance than supervised fine-tuning, despite various papers showing that modern instruction-following models can achieve similar results to supervised fine-tuning in a zero-shot setting. Despite such variances, our methodology consistently yields superior results compared to other approaches.

8 Acknowledgments

PR is a member of the Data and Marketing Insights research unit of the Bocconi Institute for Data Science and Analysis, and is supported by a MUR FARE 2020 initiative under grant agreement Prot. R20YSMBZ8S (INDOMITA). This project has been funded, in part, by DARPA under contract HR00112290106 and the Army Research Laboratory under contract W911NF-23-2-0183. Also, this research/project is supported by Ministry of Education, Singapore, under its Academic Research Fund (AcRF) Tier 2. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of the Ministry of Education, Singapore.

References

Cem Anil, Yuhuai Wu, Anders Johan Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Venkatesh Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. 2022. Exploring length generalization in large language models. In *Advances in Neural Information Processing Systems*.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Luke Breitfeller, Emily Ahn, David Jurgens, and Yulia Tsvetkov. 2019. Finding microaggressions in the wild: A case for locating elusive phenomena in social media posts. 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 1664–1674, Hong Kong, China. Association for Computational Linguistics.

Harrison Chase. 2022. LangChain.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

David Dohan, Winnie Xu, Aitor Lewkowycz, Jacob Austin, David Bieber, Raphael Gontijo Lopes, Yuhuai Wu, Henryk Michalewski, Rif A Saurous, Jascha Sohl-Dickstein, et al. 2022. Language model cascades. *arXiv preprint arXiv:2207.10342*.

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

Yuwei Fang, Shuohang Wang, Yichong Xu, Ruochen Xu, Siqi Sun, Chenguang Zhu, and Michael Zeng. 2022. Leveraging knowledge in multilingual commonsense reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3237–3246, Dublin, Ireland. Association for Computational Linguistics.

Antigoni Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Twelfth International AAAI Conference on Web and Social Media*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*.

Hongyu Gong, Alberto Valido, Katherine M Ingram, Giulia Fanti, Suma Bhat, and Dorothy L Espelage. 2021. Abusive language detection in heterogeneous contexts: Dataset collection and the role of supervised attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14804–14812.

Xiaochuang Han and Yulia Tsvetkov. 2020. Fortifying toxic speech detectors against veiled toxicity. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7732–7739, Online. Association for Computational Linguistics.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 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.

Fan Huang, Haewoon Kwak, and Jisun An. 2023. Is chatgpt better than human annotators? potential and limitations of chatgpt in explaining implicit hate speech. *arXiv preprint arXiv:2302.07736*.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

David Jurgens, Libby Hemphill, and Eshwar Chandrasekharan. 2019. A just and comprehensive strategy for using NLP to address online abuse. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3658–3666, Florence, Italy. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Hannah Kirk, Bertie Vidgen, Paul Rottger, Tristan Thrush, and Scott Hale. 2022. Hatemoji: A test suite and adversarially-generated dataset for benchmarking and detecting emoji-based hate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1352–1368, Seattle, United States. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*.

Dong-Ho Lee, Akshen Kadakia, Brihi Joshi, Aaron Chan, Ziyi Liu, Kiran Narahari, Takashi Shibuya, Ryosuke Mitani, Toshiyuki Sekiya, Jay Pujara, et al. 2022. Xmd: An end-to-end framework for interactive explanation-based debugging of nlp models. *arXiv preprint arXiv:2210.16978*.

Alyssa Lees, Daniel Borkan, Ian Kivlichan, Jorge Nario, and Tesh Goyal. 2021. Capturing covertly toxic speech via crowdsourcing. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 14–20, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jingyuan Wang, Jian-Yun Nie, and Ji-Rong Wen. 2023. The web can be your oyster for improving large language models. *arXiv preprint arXiv:2305.10998*.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Nelson F Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. *arXiv preprint arXiv:2304.09848*.

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. *arXiv preprint arXiv:1907.11692*.

Stefano Menini, Alessio Palmero Aprosio, and Sara Tonelli. 2021. Abuse is contextual, what about nlp? the role of context in abusive language annotation and detection. *arXiv preprint arXiv:2103.14916*.

Jihyung Moon, Dong-Ho Lee, Hyundong Cho, Woojeong Jin, Chan Young Park, Minwoo Kim, Jonathan May, Jay Pujara, and Sungjoon Park. 2023. Analyzing norm violations in live-stream chat. *arXiv preprint arXiv:2305.10731*.

Nicolas Ocampo, Ekaterina Sviridova, Elena Cabrio, and Serena Villata. 2023. An in-depth analysis of

implicit and subtle hate speech messages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1997–2013, Dubrovnik, Croatia. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

John Pavlopoulos, Jeffrey Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity detection: Does context really matter? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4296–4305, Online. Association for Computational Linguistics.

Patricia Rossini. 2022. Beyond incivility: Understanding patterns of uncivil and intolerant discourse in online political talk. *Communication Research*, 49(3):399–425.

Paul Rottger, Bertie Vidgen, Dirk Hovy, and Janet Pierrehumbert. 2022. Two contrasting data annotation paradigms for subjective NLP tasks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–190, Seattle, United States. Association for Computational Linguistics.

Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet Pierrehumbert. 2021. HateCheck: Functional tests for hate speech detection models. 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 41–58, Online. Association for Computational Linguistics.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2022. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5884–5906, Seattle, United States. Association for Computational Linguistics.

Abulhair Saparov and He He. 2023. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations*.

Devendra Singh, Siva Reddy, Will Hamilton, Chris Dyer, and Dani Yogatama. 2021. End-to-end training of multi-document reader and retriever for open-domain question answering. *Advances in Neural Information Processing Systems*, 34:25968–25981.

Jherez Taylor, Melvyn Peignon, and Yi-Shin Chen. 2017. Surfacing contextual hate speech words within social media. *arXiv preprint arXiv:1711.10093*.

Han Wang, Ming Shan Hee, Md Rabiul Awal, Kenny Tsu Wei Choo, and Roy Ka-Wei Lee. 2023a. Evaluating gpt-3 generated explanations for hateful content moderation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 6255–6263.

PeiFeng Wang, Aaron Chan, Filip Ilievski, Muhao Chen, and Xiang Ren. 2023b. PINTO: Faithful language reasoning using prompt-generated rationales. In *The Eleventh International Conference on Learning Representations*.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Alexandros Xenos, John Pavlopoulos, and Ion Androutsopoulos. 2021. Context sensitivity estimation in toxicity detection. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 140–145, Online. Association for Computational Linguistics.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2023. Generate rather than retrieve: Large language models are strong context generators. In *The Eleventh International Conference on Learning Representations*.

Pei Zhou, Hyundong Cho, Pegah Jandaghi, Dong-Ho Lee, Bill Yuchen Lin, Jay Pujara, and Xiang Ren. 2022. Reflect, not reflex: Inference-based common ground improves dialogue response quality. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10450–10468, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Appendix

A.1 Dataset Details

Dataset statistics and its corresponding evaluation metrics are presented in Table 4. It is important to note that the label distribution for Latent Hatred (ElSherief et al., 2021) is 34% positive and 66% negative. The maximum random accuracy for this distribution would be approximately 66, while the maximum random binary F1 score would be around 50.75.

A.2 Implementation Details

A.2.1 Zero-shot LLM Inference

In this experiments, we use `gpt-3.5-turbo` language model, as of June 2023, with specific settings of $temperature$ set to 0 and top_p set to 1, which enables us to produce deterministic output for all the baselines. Prompts used for this evaluation are in Table 6. Despite instructing the model to solely consider responses with 'yes' and 'no', it sometimes generates unclear responses in complex cases due to the internal parametric policy that is trained with human preferences (Ouyang et al., 2022). In order to address such scenarios, we opt to only consider responses featuring 'yes' as a positive prediction, assuming it carries significant confidence in its prediction. For the evaluation metrics, we consider binary F1 as shown in Table 4. The implementation of our two-step approach incurs a higher cost compared to the one-step prediction method that doesn't utilize reference. However, this increased cost is relatively insignificant as it amounts to less than $20 for all our experiments.

A.2.2 Supervised Training

We use `BERT-base` (Devlin et al., 2019) and `RoBERTa-large` (Liu et al., 2019) for the experiments. When conducting fine-tuning experiments, we implement the Adam optimizer with a maximum sequence length of 256. In each experiment, we conduct a search for the optimal learning rate amongst [2e-5, 1e-5, 5e-6, 3e-6], and batch size amongst [4, 8, 16]. Full data statistics are presented in Table 4.

A.3 Prompts used in the paper

We present prompts used in prediction for zero-shot inference (Table 5) and generating interpretation (Table 6).

Dataset	# Train	# Dev	# Test	Metric
Latent Hatred (ElSherief et al., 2021)	12,234	4,078	4,079	Binary F1
Hatemoji (Kirk et al., 2022)	-	-	3,930	Binary F1

Table 4: **Datasets** used in the paper. Report train and development only for Latent Hatred which we show supervised fine-tuning results.

Method	Instruction
Direct	`Given Post: '{`$text$`}'` `Answer yes or no for whether the post may contain implicit hate.`

Chain-of-Thought	`Given Post: '{`$text$`}'`
	`Answer yes or no for whether the post may contain implicit hate.`
	`Let's think step by step.`

Reference	`Given Post: '{`$text$`}'`
	`Passage: '{`$passage$`}'`
	`Refer to the passage above,`
	`Answer yes or no for whether the post may contain implicit hate.`

Table 5: **Prompts used for prediction**

Reference	Instruction
Implication	`Given Post: '{`$text$`}'` `Extract all the implied meaning of a post.`
Sentiment	`Given Post: '{`$text$`}'` `Identify the sentiment of a post.`
Irony	`Given Post: '{`$text$`}'` `Identify whether there is irony or sarcasm with yes/no and if there is, explain it.`
Humor	`Given Post: '{`$text$`}'` `Identify if it contains black humor and if so explain it.`
Rhetoric	`Given Post: '{`$text$`}'` `Identify if it contains a rhetorical question and if so explain why it is one.`

Table 6: **Prompts used for parametric reference generation**