Towards Generalizable Generic Harmful Speech Datasets for Implicit Hate Speech Detection

Saad Almohaimeed[1,2] **Saleh Almohaimeed**[1,3] **Damla Turgut**[1] **Ladislau Bölöni**[1]
[1] Dept. of Computer Science, University of Central Florida, Orlando, Florida
[2] Dept. of Digital Transformation, Institute of Public Administration, Riyadh
[3] Dept. of Computer Science, King Saud University, Riyadh

Abstract

Implicit hate speech has increasingly been recognized as a significant issue for social media platforms. While much of the research has traditionally focused on harmful speech in general, the need for generalizable techniques to detect veiled and subtle forms of hate has become increasingly pressing. Based on lexicon analysis, we hypothesize that implicit hate speech is already present in publicly available harmful speech datasets but may not have been explicitly recognized or labeled by annotators. Additionally, crowdsourced datasets are prone to mislabeling due to the complexity of the task and often influenced by annotators' subjective interpretations. In this paper, we propose an approach to address the detection of implicit hate speech and enhance generalizability across diverse datasets by leveraging existing harmful speech datasets. Our method comprises three key components: influential sample identification, reannotation, and augmentation using Llama-3 70B and GPT-4o. Experimental results demonstrate the effectiveness of our approach in improving implicit hate detection, achieving a +12.9-point F1 score improvement compared to the baseline.

1 Introduction

The field of harmful speech classification has garnered significant attention, with extensive research addressing various aspects of this phenomenon. Several studies focus on general hate speech, such as (Davidson et al., 2017), (Zampieri et al., 2019), (Mathew et al., 2021). Others have delved into specific forms of hate speech, including works by (Waseem and Hovy, 2016), (Founta et al., 2018), and (Ousidhoum et al., 2019). For clarity, these datasets will be referred to as generic datasets throughout this paper.

While many of these datasets include annotated examples of implicit hate speech, the reliance on crowdsourced annotators has introduced variability in labeling, with some annotators identifying implicit hate as harmful, while others do not. In contrast, publicly available datasets explicitly focused on implicit hate speech (specialized datasets) are far fewer than their generic counterparts. Leveraging the extensive data available in generic datasets and reformatting them to enhance generalizability in implicit hate detection presents a promising opportunity.

To explore this, we analyzed four generic datasets—Davidson (Davidson et al., 2017), HateXplain (Mathew et al., 2021), Waseem (Waseem and Hovy, 2016), and Founta (Founta et al., 2018)—using an offensive language lexicon developed by (Almohaimeed et al., 2024). This lexicon, comprising 1.8k offensive terms along with common obfuscated variants of the same term (e.g., substituting ' s ' with ' $ ' or the letter ' o ' with ' 0 '). Our analysis approach uses exact string matching between the dataset's samples and the lexicon entries to calculate the proportion of positive samples (annotated as harmful) that were free of offensive language. Such samples were presumed to indicate implicit hate. The results revealed that the percentage of positive samples free of offensive language was 71.7%, 14%, 36.2%, and 20.8% for Waseem, Davidson, Founta, and HateXplain, respectively. This analysis is contingent on the lexicon's comprehensiveness; some rare offensive terms may not be included, potentially classifying certain offensive samples as implicit hate. Given these findings, we propose an approach to guide models trained on generic datasets toward better generalizability for implicit hate detection while preserving their ability to identify explicit harmful speech.

Harmful speech, as defined in this paper, encompasses any expression that includes explicit hate, implicit hate, offensive language, sexism, racism or abusive speech. Furthermore, implicit hate is a subtype of harmful speech, characterized by hate conveyed in a veiled or subtle manner.

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1582–1592
December 20-24, 2025 ©2025 Association for Computational Linguistics

The key contributions of this paper are as follows:

- Introduce a novel approach to generalize high-level (general-purpose) datasets to specialized classes that exist within these datasets but lack explicit annotations.
- Develop a trusted samples dataset comprising 500 samples, designed to serve as a benchmark for evaluating various types of harmful speech.
- Apply the proposed generalization approach to adapt harmful speech datasets for the task of implicit hate classification.
- Demonstrate the utility of influential sample identification by training classifiers using three proposed configurations across four different hate speech datasets and evaluating their performance on seven datasets, using cross-dataset settings.

2 Related Work

2.1 Generalizable Implicit Hate Classifier

There are several research studies that aimed to generalize implicit hate datasets in cross-dataset evaluation settings. Some of them proposed techniques to push positive samples towards their corresponding implications along with augmentation techniques (Kim et al., 2022), bring the encoding of the explicit and implicit hate samples that share the same target close to each other (Ocampo et al., 2023), proposed a pre-trained model on ToxiGen (Hartvigsen et al., 2022), integrated with prompting techniques (Kim et al., 2023), push the embeddings of positive samples to the closest positive cluster (Almohaimeed et al., 2025) or push samples that share the same semantics into the same embedding cluster (Ahn et al., 2024). All these studies used a limited number of implicit hate datasets, such as ToxiGen (Hartvigsen et al., 2022), IHC (ElSherief et al., 2021), DYNA (Vidgen et al., 2021), SBIC (Sap et al., 2020) for training implicit hate detection models.

2.2 Identification of Influential and Noisy Samples

Most approaches for the identification of influential samples in machine learning rely on the loss function (Koh and Liang, 2017; Pruthi et al., 2020; Jinadu and Ding, 2024). These methods typically aim to mitigate the impact of influential samples during training, by either reweighting their loss or by removing them from the training dataset.

(Jinadu and Ding, 2024) proposed a method for detecting and correcting mislabeled samples in machine learning. Their approach relies on loss correction within a multi-task learning framework, including a case study focusing on a hate speech dataset. The core idea behind multi-task learning in their methodology is to separate the predictions for each label based on the perspectives of individual annotators. The authors demonstrated the effectiveness of their approach, reporting an approximate 10-point improvement in the F1 score. This improvement was observed both with and without noise injection, where baseline performance dropped significantly under noise, thereby highlighting the robustness of their method.

(Liu et al., 2020) observed that machine learning models tend to learn from correctly labeled data during early epochs but gradually begin to incorporate noisy labels in later epochs. To address this issue, they proposed a method that combines a regularization technique with a semi-supervised model. Their approach estimates the probability of the target label, enabling better generalization by avoiding overfitting and the memorization of noisy labels. Unlike early-stopping techniques that simply monitor the model during initial stages and halt training prematurely, their method emphasizes sustained learning without overfitting. Experimental results on CIFAR-10 and CIFAR-100 demonstrated that their approach achieves performance comparable to SOTA methods.

(Pruthi et al., 2020) introduced TracIn, a method for calculating the influence of each training sample based on the model's predictions for test data. This approach operates in a multi-model setting as it relies on the loss function and gradients. Their experiments spanned image and text classification tasks, as well as one regression task. The authors observed that mislabeled text samples consistently exhibited high loss values, and TracIn effectively identified these mislabeled samples in the early epochs, outperforming other SOTA methods in speed and efficiency. Their method demonstrated practical benefits, improving the accuracy on the CIFAR-10 dataset by 2 percentage points and the MNIST dataset by approximately 1 percentage point.

(Arazo et al., 2019) introduced dynamic bootstrapping, an enhancement of the static bootstrapping technique developed by (Reed et al., 2015), to address the challenge of noisy labels in image datasets during training. Unlike the static approach, dynamic bootstrapping adapts individually to each sample, providing a more flexible mechanism to avoid overfitting noisy labels. The authors also combined their method with a modified version of the data augmentation technique proposed by (Zhang et al., 2018), adapting it to operate dynamically for each sample. Experiments conducted on CIFAR-10, CIFAR-100, and TinyImageNet showed that dynamic bootstrapping effectively mitigated the impact of noisy labels and outperformed existing methods in generalization, leading to improved performance across these datasets.

(Han and Tsvetkov, 2020) proposed a method to enhance implicit hate classifiers without requiring large annotated datasets. Their approach utilized probing examples from the SBIC dataset (Sap et al., 2020) and employed several tracking methods to identify influential samples that contributed to misclassifications. These influential samples were then reannotated to improve the quality of the training data. The study demonstrated that identifying and reannotating mislabeled training samples significantly enhanced the model's performance. Among the methods tested, the gradient product proved to be the most effective technique for detecting influential samples.

Overall, influential sample identification research has largely focused on image classification, such as (Koh and Liang, 2017; Pruthi et al., 2020; Liu et al., 2020; Arazo et al., 2019), while only a small number of studies have explored text classification, such as (Han and Tsvetkov, 2020; Jinadu and Ding, 2024).

3 Methodology

3.1 Datasets

The scarcity of implicit hate datasets motivated us to leverage existing generic harmful speech datasets to improve generalizability for implicit hate detection. To address this, we utilized the generic datasets for training, which is the intended use of these artifacts. These datasets serve as the foundation for applying our approach to enhance the detection of implicit hate. The licenses and usage conditions of the datasets are list in Table 1. We note that while this paper addresses a critical and sensitive topic—hate speech—it does not include any offensive content or personally identifiable information. However, the datasets used in this work for training contain offensive content due to the nature of the task.

We evaluated the trained model on three specialized implicit hate datasets: IHC (ElSherief et al., 2021), OLID_IH (Caselli et al., 2020), and THOS_IH (Almohaimeed et al., 2023). Additionally, we cross-tested on the generic datasets to ensure that the performance on explicit harmful speech detection was not compromised by our approach.

Table 1: Licenses and usage conditions of the datasets used in this paper. **P** indicates datasets with no explicit license but made publicly available by the authors with a request for citation. **MIT** refers to the MIT License. **CC** denotes the Creative Commons Attribution 4.0 (CC BY 4.0) license, and **CCNC** refers to the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) license.

Dataset	**License**
Waseem (Waseem and Hovy, 2016)	P
Davidson (Davidson et al., 2017)	MIT
Founta (Founta et al., 2018)	CC
HateXplain (Mathew et al., 2021)	MIT
IHC (ElSherief et al., 2021)	MIT
OLID_IH (Caselli et al., 2020)	CCNC
THOS_IH (Almohaimeed et al., 2023)	CCNC

As a note, the Founta dataset as used here is shorter than its original published size from Founta et al. (Founta et al., 2018) due to data unavailability while fetching them using their published IDs through the Twitter's (X) API. Approximately 35k samples were removed as a result of platform policies or author decisions. Additionally, the labels in all datasets were unified into binary classes (normal or harmful) for generic datasets and (not implicit hate or implicit hate) for specialized datasets following the standardization introduced in (Almohaimeed et al., 2024).

3.2 Trusted Samples Dataset

Our objective is to create a dataset that is significantly smaller than typical datasets but annotated with a level of effort and consideration that would not be feasible for a larger dataset. In practice, 500 carefully curated samples are sufficient for evaluating the model. Expanding the TSD would substantially increase the burden on human annotators, making it harder to maintain high-quality labels and raising the risk of inconsistencies, par-

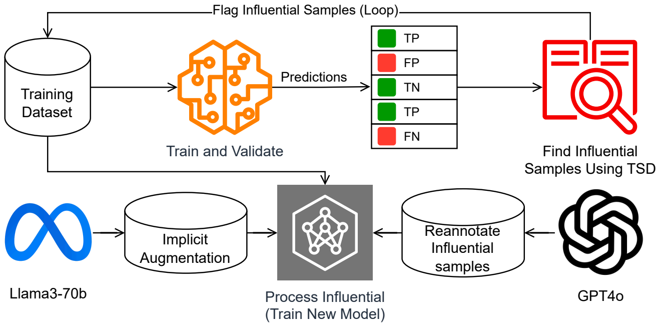

Flag Influential Samples (Loop)
Training Dataset
Predictions
🟩 TP
🟥 FP
🟩 TN
🟩 TP
🟥 FN
Train and Validate
Find Influential Samples Using TSD
Llama3-70b
Implicit Augmentation
Process Influential (Train New Model)
Reannotate Influential samples
GPT4o

Figure 1: The pipeline of our proposed methodology

ticularly in a study focused on subtle phenomena like implicit hate. To achieve this, we define the Trusted Samples Dataset (TSD[1]) as a benchmark testing dataset. Our intention is for this dataset to serve as a guide for improving data annotation to enhance generalizability and identifying influential data within the training dataset.

To construct the dataset, GPT-4o was prompted to generate examples across three levels of harmfulness (explicit, borderline, and implicit) targeting groups defined by ethnicity, religion, country, and political affiliation. To increase diversity, less common target groups were also included. All harmful examples were merged into a single positive class (see Appendix A for details). GPT-4o was additionally prompted to produce neutral samples, including some intentionally resembling implicit hate in tone and structure. Two human experts in the field of harmful speech detection jointly reviewed all generated content and retained only samples with full agreement. The final dataset contains 250 positive (harmful) and 250 negative (neutral) samples, producing the TSD dataset. The GPT-4o generation for the required content was conducted between October 8th and 12th, 2024.

The inclusion of GPT-4o in the TSD creation process was critical for achieving a comprehensive and balanced dataset. GPT-4o contributed by identifying diverse targets that human experts might overlook and generating neutral samples that closely resemble harmful speech in structure and tone. This approach provides a robust metric for evaluation, as the TSD includes challenging examples that prevent the model from favoring a specific class (positive or negative) based solely on target presence.

3.3 Influential Sample Identification

Influential samples can be classified into two categories. The first category consists of samples that were mislabeled by the annotators, leading to discrepancies in model performance. The second category includes samples that closely resemble misclassified TSD examples but belong to the opposite class in the binary classification task (harmful vs. neutral). Our objective is twofold: to correct mislabeled samples from the first category and to augment the second category to improve the model's generalizability. As depicted in Fig. 1, influential sample identification involves finding the samples responsible for the misclassification of each TSD sample.

Let us assume $D = \{t_1, t_2, t_3, \ldots, t_n\}$ where D is the training dataset and t_i represents a text sample in D. Additionally, assume $TSD = \{gt_1, gt_2, gt_3, \ldots, g_m\}$ where TSD is the trusted samples dataset and gt_j represents a text sample in TSD.

Let M be a trained model using D, and let $\hat{y_j}$ denote M's prediction for gt_j, defined $M(\mathrm{gt}_j) = \hat{y_j}, \quad$ for $j \in \{1, 2, \ldots, m\}$. We define the set of misclassified TSD samples by M as E such that $E = \{\mathrm{gt}_j \in \mathrm{TSD} \mid \hat{y_j} \neq y_j\}$ where E includes trusted samples that the model misclassified them.

Next, let us define a cosine similarity metric as $csim(t_i, gt_j) = \frac{e_{t_i} e_{gt_i}}{\|e_{t_i}\| \cdot \|e_{gt_i}\|}$, where e represents the embedding of a given text sample produced by M. Finally, we define the top x influential samples for $\mathrm{gt}_j \in$ E as follows:

$$\text{top\_influence}(E, x) = \bigcup_{gt_j \in E} \text{Top-}x\{csim(t_i, gt_j) \mid t_i \in D, y_i = \hat{y_j}\}$$

The prerequisite of this function is where the ground-truth of the D sample y_i has the same label

[1] https://github.com/mohaimeed/TSD

of misclassified TSD sample $\hat{y_j}$.

3.4 GPT4 Annotation

Sometimes what makes a sample influential in making M misclassify a given gt_j is the fact that the sample was mislabeled by the annotators. It is difficult to engage humans in the loop of our pipeline Fig. 1 to reannotate influential samples since we are dealing with an extensive amount of data, 107.8k rows in total from the 4 generic datasets along with full training process on each loop in our pipeline. So, it is necessary to find an applicable technique to reannotate influential samples such that it will be fixed in case it was mislabeled. Despite the limitations of GPT4 to overlook the explicit and implicit hate that targets individuals or to be confused with sarcasms and opinions (Almohaimeed et al., 2024), several studies (Almohaimeed et al., 2024; Huang et al., 2023; Dönmez et al., 2024) explain the effectiveness of GPT family on identifying hate text. (Dönmez et al., 2024) studies the effectiveness and limitations of 16 LLMs on identifying the harmful content, and despite the limitations, the authors found that the GPT family (GPT3.5-turbo and GPT4) were the best closest performing LLMs to the human baseline. Taking these results into consideration, we employed GPT4o to reannotate the influential samples, and adjust the ground truth label if it does not match the GPT4o annotation.

3.5 Llama-3 Augmentation

We utilized the capabilities of open-source large language models to augment generic datasets with more implicit hate speech samples. Specifically, we employed Llama-3 8B and 70B (Dubey et al., 2024) to paraphrase explicit harmful speech samples into implicit hate speech.

Llama-3 8B struggled to perform the task effectively, often repeating the provided system and user prompts instead of generating paraphrased outputs. In contrast, Llama-3 70B demonstrated a substantial understanding of the task and consistently produced accurate paraphrases. For this augmentation process, we used the 4-bit quantization setting on an NVIDIA H100 80GB GPU. The paraphrasing of a single harmful speech sample took an average of 9.58 seconds, resulting in a total runtime of approximately 136 hours to augment 51k harmful speech samples.

In the experiments, we explored several techniques to incorporate the augmented data into model training. The first approach duplicated each row with its augmented version, assigning the same label (positive). The second approach replaced explicit harmful speech rows with their implicit counterparts. However, in both cases, the BERT model struggled to learn from the augmented data and failed to converge, even when varying model hyperparameters.

Our observations indicate that models often face difficulties when training on datasets with a high proportion of deeply implicit samples — samples that even humans may find challenging to identify as harmful. Examples include statements like *"RACE should stay in their own neighborhoods"* or *"COUNTRY people need to take more responsibility"*. While such statements may not appear harmful to some, they can be deeply offensive to individuals belonging to vulnerable groups. These challenges are compounded for small models (e.g. BERT), which may lack the capacity to discern the implicit harmfulness of such texts.

In our case, we limited augmentation to the identified influential samples—a smaller subset of the dataset—which yielded significant performance improvements, as shown in Table 4.

4 Experimental Settings

For our experiments settings, we utilized BERT$_{\text{base}}$ uncased, which comprises 110 million parameters and 12 encoder layers (Kenton and Toutanova, 2019). We experimented with batch sizes of 4, 8, and 16 and learning rates ranging from 1e-6 to 1e-7. Although all experiments were run for 20 epochs, the reported results are based on the best epoch performance.

4.1 Experimental Setup

Text Preprocessing: We applied several steps to clean and standardize the input data for the training model. These steps included removing URLs, user references (e.g., @USER), and hashtags from the text. We also decomposed contractions (e.g., doesn't → does not) and eliminated extra whitespaces. These preprocessing steps were applied to ensure consistency and reduce noise in the dataset.

Generic Datasets: This category comprised four experiments using generic datasets for training. In these experiments, we focused on influential sample identification and processing. Each dataset went through multiple full training loops. During each loop, influential samples were identified, removed from the dataset, and then the model was retrained. The number of loops required to achieve the best results varied across datasets. Table 2 shows the number of loops and the number of dropped samples for each dataset. The number of dropped samples represents the total count of the top x samples for each misclassified gt_i. It is important to note that that the top x samples for gt_a and gt_b may sometimes overlap. For Founta, we chose to drop the top 20 samples for each misclassified gt_j sample per loop instead of 10, given its larger dataset size relative to the other generic datasets.

Table 2: Influential Sample Identification Results of the Generic Datasets

Dataset	# Top	# Loops	Original Size	# Influential Samples
Waseem	10	16	16,907	8,343
Davidson	10	3	24,783	1,434
Founta	20	13	45,982	12,770
HateXplain	10	7	20,148	2,328

Specialized Datasets: This category involved three experiments where the model was trained on specialized datasets (i.e. implicit hate datasets) and tested across other specialized datasets. The objective was to compare the effectiveness of generalizing from generic datasets toward implicit hate detection versus using datasets explicitly specialized for implicit hate.

Testing: For each test across the seven datasets, the results presented in Tables 3, 4 and 5 represent the average performance across five run with different random seeds. Each run used balanced random samples consisting of 500 positive and 500 negative examples from the test set. This approach ensured consistency and robustness for evaluating the results across all experiments.

4.2 Metrics

The metrics used in all experiments were the F1-Micro score and the Recall. While the F1 score is a more comprehensive metric as it incorporates Recall in its calculation, Recall was also included as an additional metric to ensure fairness when comparing the performance of generic datasets (Table 4) against specialized datasets (Table 5). This consideration stems from the inherent differences in the annotation approaches: generic datasets label all harmful speech samples as positive, whereas specialized datasets label only implicit hate samples as positive. Consequently, other types of harmful speech (e.g., explicit hate and offensive language) are annotated as negative in specialized datasets, alongside neutral speech. As a result, when a model trained on generic datasets is tested on specialized datasets, it is likely to predict many explicit harmful speech samples as positive, resulting in a higher number of False Positives (FP). This discrepancy occurs because specialized datasets annotate such samples as negative, focusing solely on implicit hate. Given this context, Recall is particularly suitable as it measures the True Positive (TP) rate, capturing the proportion of correct positive predictions relative to the total number of actual positive samples. In our case, Recall provides a reliable measure of the model's ability to identify implicit hate samples within the specialized datasets, providing a fair and meaningful comparison. However, we also include the F1 score in Tables 4 and 5 to ensure a balanced view of the model's performance without overemphasizing Recall at the expense of Precision.

5 Experiments and Results

We defined three approaches for conducting the experiments and compare them against the baseline (training with the original dataset). The first approach involves training a generic model after removing all influential samples from the training dataset (see Table 2). The second approach involving reannotating the influential samples within each training dataset using GPT-4o, as detailed in Section 3.4. The third approach involves augmenting the influential samples with paraphrased implicit versions generated by Llama-3 70B, as described in Section 3.5. For each training dataset listed in Table 3, the last row (exploration) shows results where the influential samples were reannotated in both the training and testing datasets. This was done to explore the impact of testing the model on reannotated (cleaned) datasets.

5.1 Generic Dataset as Training and Testing

In this experiment, the F1 score is more meaningful than Recall, although Recall is included for consistency across the tables. The F1 score is more relevant here because both the training and testing datasets classify any kind of harmful speech—whether explicit, implicit, or offensive—as part of the positive class. As shown in Table 3 and Fig. 2a, the performance varies depending

Table 3: Performance of generic datasets evaluated across each other. Baseline results are underlined, and the best-performing approach is highlighted in **bold**.

↓ Train Dataset	Test dataset→	Waseem		Davidson		Founta		HateXplain	
		R	F1	R	F1	R	F1	R	F1
Waseem	Original Dataset	-	-	0.743	**0.836**	0.597	0.796	0.547	**0.76**
	+ Drop Influential Samples	-	-	0.781	0.759	0.69	0.805	0.768	0.744
	+ GPT4o Influential Reannotation	-	-	0.776	0.767	0.695	0.827	0.8	0.657
	+ GPT4o + Llama3 Augmentation	-	-	**0.822**	0.759	**0.747**	**0.846**	**0.853**	0.648
	+ GPT4o on Train and Test	-	-	0.761	0.774	0.728	0.849	0.785	0.672
Davidson	Original Dataset	0.652	0.7	-	-	0.772	0.81	0.869	**0.669**
	+ Drop Influential Samples	**0.866**	0.697	-	-	**0.86**	0.783	**0.936**	0.598
	+ GPT4o Influential Reannotation	0.702	0.724	-	-	0.798	**0.832**	0.922	0.627
	+ GPT4o + Llama3 Augmentation	0.764	**0.729**	-	-	0.826	0.824	0.914	0.616
	+ GPT4o on Train and Test	0.71	0.752	-	-	0.826	0.849	0.909	0.637
Founta	Original Dataset	0.596	**0.742**	0.874	**0.852**	-	-	0.784	**0.724**
	+ Drop Influential Samples	**0.68**	0.737	**0.895**	0.805	-	-	0.813	0.717
	+ GPT4o Influential Reannotation	0.616	0.738	0.886	0.829	-	-	**0.847**	0.684
	+ GPT4o + Llama3 Augmentation	0.562	0.716	0.882	0.836	-	-	0.82	0.699
	+ GPT4o on Train and Test	0.64	0.776	0.875	0.824	-	-	0.844	0.695
HateXplain	Original Dataset	0.658	0.704	0.953	0.726	0.797	**0.831**	-	-
	+ Drop Influential Samples	0.715	**0.705**	0.953	**0.727**	0.804	**0.831**	-	-
	+ GPT4o Influential Reannotation	0.762	0.695	0.957	0.711	0.814	0.822	-	-
	+ GPT4o + Llama3 Augmentation	**0.777**	0.665	**0.961**	0.699	**0.827**	0.811	-	-
	+ GPT4o on Train and Test	0.762	0.731	0.962	0.728	0.851	0.838	-	-

on the training dataset. The $M(Waseem)$ dataset performed best in the baseline. The $M(Davidson)$ showed a preference for the second approach, achieving an average F1 score of 0.728 across the three test datasets. For the $M(Founta)$, the baseline outperformed other approaches by an average of 2 points in the F1 score. Finally, the $M(HateXplain)$ performed slightly better with the first approach, though the results were nearly identical to the baseline.

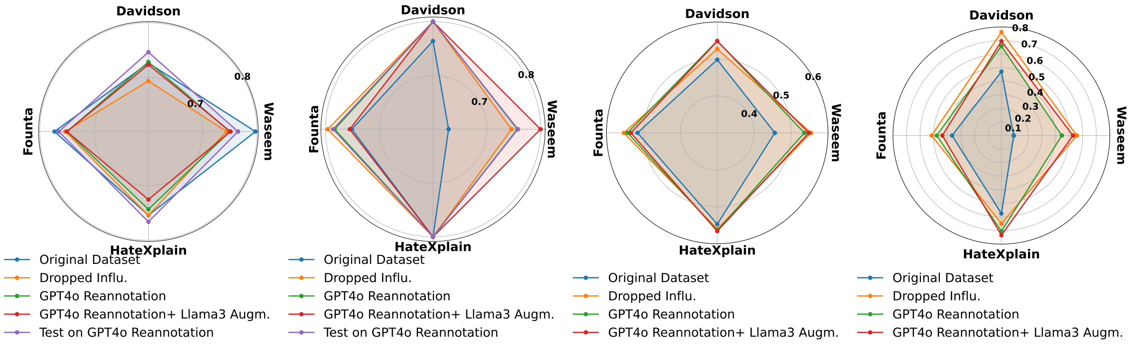
5.2 Generic Dataset as Training and Specialized as Testing

In this experiment, the Recall metric is more meaningful than the F1 score because the generic-trained model is expected to produce a high number of False Positives (FP) when tested on implicit hate datasets, where explicit harmful speech is labeled as part of the negative class. As shown in Table 4 and Fig. 2d, the models trained on Waseem, Davidson, and Founta datasets favored the first approach, achieving the best performance on implicit hate testing. In contrast, models trained on HateXplain performed best with the third approach.

Regarding model generalizability, Fig. 2c and Fig. 2d illustrate that all proposed approaches substantially improved performance compared to training on the original datasets. This demonstrates the effectiveness of the proposed methods in generalizing generic datasets for implicit hate detection in specialized datasets. Table 5 further highlights the performance of specialized datasets in cross-dataset testing scenarios. Despite the limited availability of implicit hate datasets, we evaluated their effectiveness when the model was trained and tested on the same type of dataset, comparing the results with our generalized approaches. The results indicate that $M(THOS\_IH)$ and $M(OLID\_IH)$ struggled to perform well in cross-dataset testing, whereas the $M(IHC)$ showed stronger performance. However, when examining the average Recall across tests with OLID_IH and THOS_IH, $M(Davidson)$ in the first approach and $M(HateXplain)$ in the third approach outperformed the specialized dataset's model $M(IHC)$. Additionally, as shown in Table 4, the remaining results were comparable to $M(IHC)$ performance in Table 5. In terms of the F1 score, $M(HateXplain)$ under the third approach also outperformed $M(IHC)$.

Table 4: Performance of generic datasets evaluated across specialized datasets. Baseline results are underlined, and the best-performing approach is highlighted in **bold**.

↓ Train Dataset	Test dataset→	IHC		OLID_IH		THOS_IH	
		R	F1	R	F1	R	F1
Waseem	Original Dataset	0.064	0.512	0.14	0.501	0.069	0.355
	+ Drop Influential Samples	0.64	0.588	**0.564**	**0.59**	**0.467**	**0.484**
	+ GPT4o Influential Reannotation	0.573	**0.607**	0.409	0.553	0.356	0.447
	+ GPT4o + Llama3 Augmentation	**0.661**	**0.607**	0.518	0.576	0.406	0.462
Davidson	Original Dataset	0.598	0.563	0.424	0.509	0.39	0.422
	+ Drop Influential Samples	**0.851**	0.562	**0.789**	0.55	**0.647**	0.469
	+ GPT4o Influential Reannotation	0.8	**0.607**	0.649	0.566	0.527	**0.475**
	+ GPT4o + Llama3 Augmentation	0.82	0.6	0.714	**0.571**	0.55	0.474
Founta	Original Dataset	0.459	0.586	0.378	0.546	0.256	0.415
	+ Drop Influential Samples	0.535	0.582	**0.58**	**0.594**	**0.43**	**0.482**
	+ GPT4o Influential Reannotation	**0.612**	**0.611**	0.476	0.57	0.344	0.448
	+ GPT4o + Llama3 Augmentation	0.544	0.6	0.459	0.569	0.306	0.432
HateXplain	Original Dataset	0.701	**0.62**	0.58	0.56	0.443	0.462
	+ Drop Influential Samples	0.721	0.605	0.68	0.571	0.545	0.504
	+ GPT4o Influential Reannotation	0.82	0.609	0.712	0.571	0.587	0.511
	+ GPT4o + Llama3 Augmentation	**0.824**	0.6	**0.768**	**0.577**	**0.614**	**0.521**



(a) Generic F1 Score (b) Generic Recall Score (c) Specialized F1 Score (d) Specialized Recall Score

Figure 2: Averaged Performance Metrics for training on generic datasets and evaluated over generic and specialized datasets (generic datasets in the figures represent the training dataset).

Table 5: Performance of specialized datasets evaluated across each other (for comparison with the performance of the generic datasets in Table 4).

Test Dataset → Train Dataset ↓	IHC		OLID_IH		THOS_IH	
	R	F1	R	F1	R	F1
IHC	-	-	0.659	0.557	0.712	0.529
OLID_IH	0.136	0.529	-	-	0.077	0.516
THOS_IH	0.047	0.51	0.016	0.502	-	-

5.3 Reannotated Dataset as Training and Testing

As shown in the last row of each training dataset section in Table 3, 7 out of 12 experiments achieved higher F1 scores compared to all other approaches, including the baseline. The results show that reannotating Davidson and HateXplain on testing led to a decrease in performance. In contrast, reannotating Waseem and Founta as testing datasets resulted in improved performance in all experiments, outperforming other approaches, including the baseline. This discrepancy raises important questions. Given that Waseem and Founta have a high proportion of influential samples, as shown in Table 2, it is possible that these datasets initially contained significant noise, which was corrected through relabeling. Conversely, this result may suggest that HateXplain and Davidson require additional training loops, as illustrated in our pipeline (Fig. 1), to identify and correct more influential samples for further improvements.

From a different perspective, when comparing the exploration approach, in which GPT-4o reannotation was applied to both training and testing data, with the second approach, where GPT-4o reannotation was applied only to the training data, the exploration approach yielded better performance in all experiments. The only exception occured when the model was trained with Founta and tested with Davidson, where the second approach achieved an F1 score of 0.829, slightly higher than the 0.824 score for the exploration approach.

6 Conclusion

In this paper, we proposed an approach to generalize generic datasets toward a subtle and previously unannotated class by leveraging a trusted samples dataset generated with GPT-4o under the guidance and curation of two experts. The approach involves identifying influential samples in the training data and applying various configurations, such as removing influential samples, GPT-4o reannotation, and Llama-3 augmentation. To evaluate the performance of our proposed approach, we conducted experiments using seven datasets of harmful speech. Among these, four are generic datasets focusing on hate speech and offensive language, while three are specialized datasets on implicit hate speech. Our results demonstrate the effectiveness of the proposed approach to generalize the explicit hate datasets to classify implicit hate samples, achieving a 12.9 point improvement in F1 score on specialized datasets while maintaining comparable performance on the generic datasets.

Limitations

In our second and third approaches, where we engaged the LLM to determine whether a given influential sample is mislabeled, the solution remains suboptimal. As observed in previous research (Almohaimeed et al., 2024), (Dönmez et al., 2024), LLMs have not yet reached the level of human experts in accurately identifying harmful content, particularly in its implicit form. This limitation may lead to the mislabeling of critical data, posing a challenge to the reliability of the model.

Additionally, in our methodology pipeline, the selection of the best-performing version of the training dataset after removing a set of influential samples is not automated. Instead, the choice was made based on our manual observation of the optimal loop results. Developing an approach to systematically determine whether a given version yields the best results would enable a fully automated pipeline. Such an advancement could be beneficial for future research and facilitate the development of tools for both academic and production settings.

References

Hyeseon Ahn, Youngwook Kim, Jungin Kim, and Yo-Sub Han. 2024. SharedCon: Implicit hate speech detection using shared semantics. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics.

Saad Almohaimeed, Saleh Almohaimeed, and Ladislau Bölöni. 2024. Transfer learning and lexicon-based approaches for implicit hate speech detection: A comparative study of human and GPT-4 annotation. In *Proc. of 2024 IEEE 18th Int. Conf. on Semantic Computing (ICSC-2024)*, pages 142–147.

Saad Almohaimeed, Saleh Almohaimeed, Ashfaq Ali Shafin, Bogdan Carbunar, and Ladislau Bölöni. 2023. THOS: A benchmark dataset for targeted hate and offensive speech. In *Proc. of Data-centric Machine Learning Research (DMLR) Workshop at ICML 2023*.

Saad Almohaimeed, Saleh Almohaimeed, Damla Turgut, and Ladislau Bölöni. 2025. Closest positive cluster loss: Improving the generalization of implicit hate speech classifiers across social media datasets. In *Proc. of IEEE Int. Conf. on Communications (ICC-2025)*.

Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. 2019. Unsupervised label noise modeling and loss correction. In *Proc. of Int. Conf. on Machine Learning (ICML-2019)*, pages 312–321.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. 2020. I feel offended, don't be abusive! Implicit/explicit messages in offensive and abusive language. In *Proc. of the Twelfth Language Resources and Evaluation Conf.*, pages 6193–6202.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proc. of the Int. AAAI Conf. on Web and Social Media (ICWSM-2017)*, volume 11, pages 512–515.

Esra Dönmez, Thang Vu, and Agnieszka Falenska. 2024. Please note that I'm just an AI: Analysis of behavior patterns of LLMs in (non-) offensive speech identification. In *Proc. of the 2024 Conf. on Empirical Methods in Natural Language Processing EMNLP-2024*, pages 18340–18357.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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 *Proc. of the Conf. on Empirical Methods in Natural Language Processing (EMNLP-2021)*, pages 345–363.

Antigoni 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 *Proc. of the Int. AAAI Conf. on Web and Social Media (ICWSM-2018)*, volume 12.

Xiaochuang Han and Yulia Tsvetkov. 2020. Fortifying toxic speech detectors against veiled toxicity. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing (EMNLP-2020)*, pages 7732–7739.

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 *Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (ACL-2022)*.

Fan Huang, Haewoon Kwak, and Jisun An. 2023. Is ChatGPT better than human annotators? Potential and limitations of ChatGPT in explaining implicit hate speech. In *Companion Proc. of the ACM Web Conference 2023*, page 294–297.

Uthman Jinadu and Yi Ding. 2024. Noise correction on subjective datasets. In *Proc. of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL-2024)*, pages 5385–5395.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of the 2019 Conf. of the North American chapter of the Association for Computational Linguistics (NAACL-2019)*, pages 4171–4186.

Youngwook Kim, Shinwoo Park, and Yo-Sub Han. 2022. Generalizable implicit hate speech detection using contrastive learning. In *Proc. of the 29th Int. Conf. on Computational Linguistics (COOLING-2022)*, pages 6667–6679.

Youngwook Kim, Shinwoo Park, Youngsoo Namgoong, and Yo-Sub Han. 2023. ConPrompt: Pre-training a language model with machine-generated data for implicit hate speech detection. In *Proc. of Conf. on Empirical Methods in Natural Language Processing (EMNLP 2023)*.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proc. of Int. Conf. on Machine Learning (ICML-2017)*, pages 1885–1894.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. 2020. Early-learning regularization prevents memorization of noisy labels. *Advances in Neural Information Processing Systems (NeurIPS-2020)*, 33:20331–20342.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. HateXplain: A benchmark dataset for explainable hate speech detection. In *Proc. of the AAAI Conf. on Artificial Intelligence (AAAI-2021)*, volume 35, pages 14867–14875.

Nicolás Benjamín Ocampo, Elena Cabrio, and Serena Villata. 2023. Unmasking the hidden meaning: Bridging implicit and explicit hate speech embedding representations. In *Proc. of Conf. on Empirical Methods in Natural Language Processing (EMNLP 2023)*.

Nedjma Ousidhoum, Zizheng Lin, Hongming Zhang, Yangqiu Song, and Dit-Yan Yeung. 2019. Multilingual and multi-aspect hate speech analysis. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing (EMNLP-IJCNLP-2019)*, pages 4675–4684.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems (NeurIPS-2020)*, 33:19920–19930.

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Alexander Rabinovich. 2015. Training deep neural networks on noisy labels with bootstrapping. In *Proc. of Int. Conf. on Learning Representations (ICLR-2015)*.

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

Bertie Vidgen, Tristan Thrush, Zeerak Waseem, and Douwe Kiela. 2021. Learning from the worst: Dynamically generated datasets to improve online hate detection. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing (EMNLP-IJCNLP-2021)*.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter. In *Proc. of the NAACL Student Research Workshop*, pages 88–93.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. In *Proc. of the 2019 Conf. of the North American chapter of the Association for Computational Linguistics (NAACL-2019)*, pages 1415–1420.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *Proc. of Int. Conf. on Learning Representations (ICLR-2018)*.

7 Appendix A

The process of GPT-4o generation involved identifying core vulnerable groups (ethnicity, country, religion, political affiliation) and prompting GPT-4o to propose additional, less common targets to increase diversity. The degree of explicitness (explicit, borderline, or implicit) of the harmful content was also defined during the prompting process and is described as follows:

- Explicit hate: clearly overt expressions.
  Examples: “RELIGION are criminal” or “ETHNICITY are violent m**f who should all be in prison.”

- Borderline hate: explicit to some readers, but not to others.
  Examples: “ETHNICITY are ruining our culture” or “COUNTRY are always looking for handouts.”

- Implicit hate: subtle or veiled, difficult to detect for both LLMs and some human annotators.
  Example: “RELIGION are often too controlling of global financial systems.”

These examples are real samples from the TSD. The three categories were used only during data creation; in the final dataset, they were unified into a single positive class representing harmful content.