

Article

PEMTCL: A Prompt-Enhanced Multi-Task Contrastive Learning Framework for Fine-Grained Toxic Language Detection

Shan Jin ¹ , Xiaochao Fan ^{1,2,*}, Zhenzhen He ¹ and Peng Chen ¹

¹ College of Computer Science and Technology, Xinjiang Normal University, Urumqi 830054, China; 107622024210513@stu.xjnu.edu.cn (S.J.); hezhzhenzhen@xjnu.edu.cn (Z.H.); chenpeng718@126.com (P.C.)

² Xinjiang Engineering Technology Research Center for Smart Education and Applications, Urumqi 830054, China

* Correspondence: fxc1982@xjnu.edu.cn

Abstract

Fine-grained toxic language detection aims to assess the harmfulness, expression style, and attack target of textual content, supporting refined content moderation on online platforms. Most existing methods model each subtask independently, failing to exploit semantic correlations among tasks, while conventional contrastive learning is prone to introducing false negative samples under multi-task, multi-label settings. To address these limitations, we propose a Prompt-Enhanced Multi-Task Contrastive Learning framework (PEMTCL). The framework employs a dynamic prompt generation mechanism to construct instance-level soft prompts, enabling the encoder to adaptively capture fine-grained semantic cues required by different subtasks. It then jointly optimizes four subtasks—toxicity identification, toxic type discrimination, expression type detection, and targeted group detection—achieving inter-task knowledge complementarity within a shared representation space. Furthermore, a label-aware multi-head contrastive learning module leverages multi-task label relationships for reliable negative sample selection and incorporates a sample reweighting mechanism, effectively mitigating false negatives and class imbalance. Experiments on two Chinese datasets, ToxiCN and FG-COLD, demonstrate that PEMTCL achieves the best performance on most subtasks, with notable improvements on fine-grained tasks. Ablation studies, training data scale analysis, and case studies further validate the effectiveness and robustness of the proposed model.

Keywords: fine-grained toxic language detection; multi-task learning; contrastive learning; prompt learning; label-aware negative sampling



Academic Editor: Yazhou Zhang

Received: 28 June 2026

Revised: 24 July 2026

Accepted: 6 August 2026

Published: 7 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

With the rapid proliferation of social media, toxic language in user-generated content has increased significantly, becoming one of the critical challenges in cyberspace governance [1,2]. Toxic language typically targets specific groups through either explicit or implicit expressions [3], conveying offensive or hateful content that can readily provoke social polarization and psychological harm [4]. Conducting multi-dimensional identification of toxic language facilitates a more comprehensive characterization of its toxicity level, expression style, and attack target, while also providing finer-grained evidence for platform content moderation and tiered response.

Toxic language detection typically involves multiple tasks, including determining whether a statement is toxic, identifying the toxicity type, distinguishing between explicit

and implicit expressions, and recognizing hate targets. Lu et al. [5] constructed ToxiCN, a fine-grained Chinese toxic language dataset, and independently modeled the toxicity, type, expression style, and hate target of utterances using BERT-based methods. Pan et al. [6] built a fine-grained toxic language dataset in Spanish, treating speech type, attack target, target group, and intensity as four related subtasks, and adopted an mBART-based multi-task learning framework, demonstrating the feasibility of jointly learning fine-grained toxic language tasks. However, existing methods still commonly model multiple fine-grained tasks independently (as illustrated in Figure 1), or merely adopt simple shared structures, making it difficult to fully exploit inter-task correlations and complementary information, thereby limiting the model's comprehensive recognition capability for complex fine-grained toxic samples.

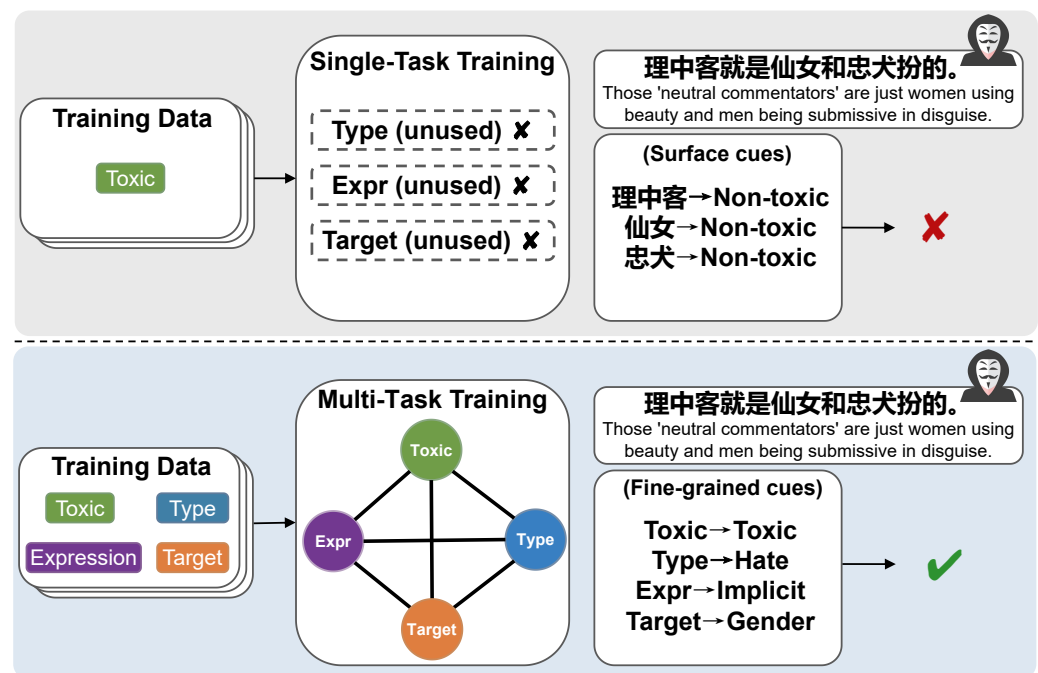


Figure 1. Illustration of single-task versus multi-task training. The checkmark denotes a correct prediction, whereas the cross denotes an incorrect prediction.

Contrastive learning has been widely employed to enhance model discriminability in toxic language detection, particularly for implicit expressions [7]. Dehghan and Yanikoğlu [8] proposed a dual contrastive learning framework that incorporates paralinguistic features for multi-domain hate speech detection. Rehman et al. [9] proposed a two-stage contrastive learning framework for implicit hate speech detection in videos, extending contrastive constraints beyond textual modality to improve cross-modal representation transferability. However, existing contrastive learning methods are predominantly designed for single-task or single-label scenarios. Under multi-task, multi-label settings, how to effectively partition negative samples based on label relationships across different tasks, and further attend to hard negative samples that are semantically similar to the anchor yet carry different labels, remains an open research question.

To address the aforementioned issues, we propose a Prompt-Enhanced Multi-Task Contrastive Learning framework for fine-grained toxic language detection (PEMTCL). The framework achieves task-adaptive input augmentation through dynamic prompt generation, enabling the encoder to capture fine-grained semantic cues. Under a multi-task learning paradigm, it jointly models four subtasks—toxicity identification, toxic type discrimination, expression type detection, and targeted group detection—fully leveraging complementary information across subtasks. Meanwhile, label-aware multi-head con-

trastive learning is introduced to exploit multi-task label relationships, enhancing the discriminability for implicit toxicity and highly confusable samples. The framework jointly optimizes multi-task losses and contrastive losses, improving the model's recognition performance for complex toxic expressions and fine-grained semantic distinctions. The main contributions of this paper are summarized as follows:

- To address the semantic correlation loss and insufficient task adaptation caused by independently modeling fine-grained subtasks in toxic language detection, we propose a unified learning framework that integrates dynamic prompt generation with multi-task joint optimization. This framework achieves fine-grained joint modeling of toxic semantics at the task-structure level, enhancing knowledge sharing and complementarity across different toxicity attributes.
- To address the problem that conventional contrastive learning tends to introduce false negative samples and neglects fine-grained semantic correlations under multi-label, multi-task settings in toxic text detection, we propose a label-aware multi-head contrastive learning method. This method exploits multi-dimensional label relationships at the representation space level to construct more reliable positive and negative sample constraints, mitigating representation learning biases caused by false negative samples and class imbalance.
- Experimental results on public datasets demonstrate that the proposed PEMTCL framework achieves the best performance across multiple fine-grained toxic language detection tasks, fully validating its effectiveness.

2. Related Work

2.1. Fine-Grained Toxic Language Detection

In recent years, research on toxic language detection has shifted from coarse-grained binary classification toward fine-grained analysis, with a focus on toxicity type, expression style, and hate target. Lu et al. [5] constructed ToxiCN, a fine-grained Chinese toxic language dataset, and proposed a multi-level annotation scheme covering toxicity type, expression style, and target group. They also designed the TKE model that integrates lexical knowledge, and experimentally validated its effectiveness in both explicit and implicit toxicity analysis. Building on this direction, Ghorbanpour et al. [10] introduced label-specific soft prompt tuning for fine-grained transfer learning in harmful content detection, further improving adaptability across heterogeneous label taxonomies. Nghiem et al. [11] constructed HateCOT, an explanation-enhanced dataset for generalizable offensive speech detection via large language models, and found that existing models struggle to generalize across coded, indirect, and context-dependent hate expressions without explicit explanatory supervision. Calabrese et al. [12] proposed a compositional generalisation framework for explainable hate speech detection, demonstrating that decomposing model reasoning into interpretable components can improve generalization to unseen target groups and alleviate over-reliance on identity terms. Mothilal et al. [13] examined argument-based consistency in large language model explanations of toxicity, finding that models with high classification performance do not necessarily exhibit good explanation consistency, a concern further echoed by recent explanation-quality evaluation frameworks [14,15].

2.2. Multi-Task Learning for Toxic Language Detection

Multi-task learning is commonly employed to simultaneously model multiple related classification objectives, enabling models to leverage complementary information across different tasks within shared representations. Pan et al. [6] constructed a fine-grained toxic language dataset in Spanish, treating speech type, attack target, target group, and intensity as four subtasks, and adopted mBART for joint modeling. The results demonstrated

that multi-task learning can achieve competitive performance in fine-grained hate speech detection. Hoang et al. [16] proposed ToXCL, which unifies implicit toxicity detection and explanation generation, and introduces target group generation and knowledge distillation to improve the detection performance for implicit toxic text. Büyükdemirci et al. [17] constructed JL-Hate, an annotated dataset for joint learning of hate speech and target detection, demonstrating that joint annotation of multiple hate dimensions helps mitigate interference in standard multi-task learning.

2.3. Contrastive Learning for Toxic Language Detection

Contrastive learning enhances model discriminability for implicit expressions and confusable samples by adjusting representation distances between samples. Kim et al. [18] proposed ImpCon, which pulls implicit hate text and its implied meanings closer within the same representation space, improving the model's generalization for implicit hate detection across datasets. Lee et al. [19] proposed AmpleHate, which amplifies attention weights to enhance versatile implicit hate detection, achieving results superior to multiple baselines on English hate speech detection datasets. Kim et al. [20] proposed ConPrompt, which leverages source prompts of machine-generated samples to construct positive sample relationships, making pre-trained models more adaptable to implicit hate speech detection. J. Kim et al. [7] proposed LAHN, combining label-aware hard negative sampling with momentum contrastive learning, directing the model to focus more on samples that are semantically similar to the anchor yet carry different labels, and achieving favorable performance under both in-domain and cross-dataset settings.

3. Problem Formulation

We formulate the four subtasks of toxic language detection as a joint prediction problem. Given an input text sequence $x = (w_1, w_2, \dots, w_n)$, where n denotes the text length, the model is required to learn a mapping function $F(\cdot)$ that maps the input text to a four-dimensional semantic tuple $F(x) \rightarrow (y_{\text{tox}}, y_{\text{type}}, y_{\text{expr}}, y_{\text{tgt}})$, where y_{tox} is a single-valued label and the others are multi-valued labels. Each dimension is defined as follows:

- **Toxicity Identification:** $y_{\text{tox}} \in \{0, 1\}$, where 0 indicates Non-Toxic and 1 indicates Toxic.
- **Toxic Type Discrimination:** $y_{\text{type}} \in \{0, 1\}^{C_{\text{type}}}$, where $C_{\text{type}} = 2$, corresponding to General Offensive Language and Hate Speech, respectively.
- **Expression Type Detection:** $y_{\text{expr}} \in \{0, 1\}^{C_{\text{expr}}}$, where $C_{\text{expr}} = 3$, corresponding to Explicitness, Implicitness, and Reporting, respectively.
- **Targeted Group Detection:** $y_{\text{tgt}} \in \{0, 1\}^{C_{\text{tgt}}}$, where $C_{\text{tgt}} = 4$, corresponding to Gender, Race, Region, and LGBTQ. This is a multi-label classification task, as a single text may simultaneously target multiple groups.

4. Methodology

4.1. Framework Overview

The proposed PEMTCL framework, as illustrated in Figure 2, consists of three main modules: dynamic prompt generation, multi-task learning, and label-aware multi-head contrastive learning.

First, to address the issue that different subtasks focus on different semantic cues, the dynamic prompt generation mechanism adaptively fuses shared source prompts with task-level priors based on the input sample to construct instance-level soft prompts, thereby enhancing the encoder's adaptability to the discriminative requirements of different subtasks. Second, considering that the four subtasks are semantically correlated yet have different output spaces, the multi-task learning module shares a pre-trained encoder and

jointly optimizes toxicity identification, toxic type discrimination, expression type detection, and targeted group detection, accommodating both coarse-grained and fine-grained toxicity features within a unified representation space. Finally, to address the difficulty of distinguishing implicit toxic samples and highly confusable negative samples, label-aware multi-head contrastive learning leverages multi-task label information to construct fine-grained positive and negative sample relationships, expanding negative sample sources through a momentum queue, filtering false negative samples based on label overlap rates, and introducing a sample reweighting mechanism to strengthen constraints on hard negative samples.

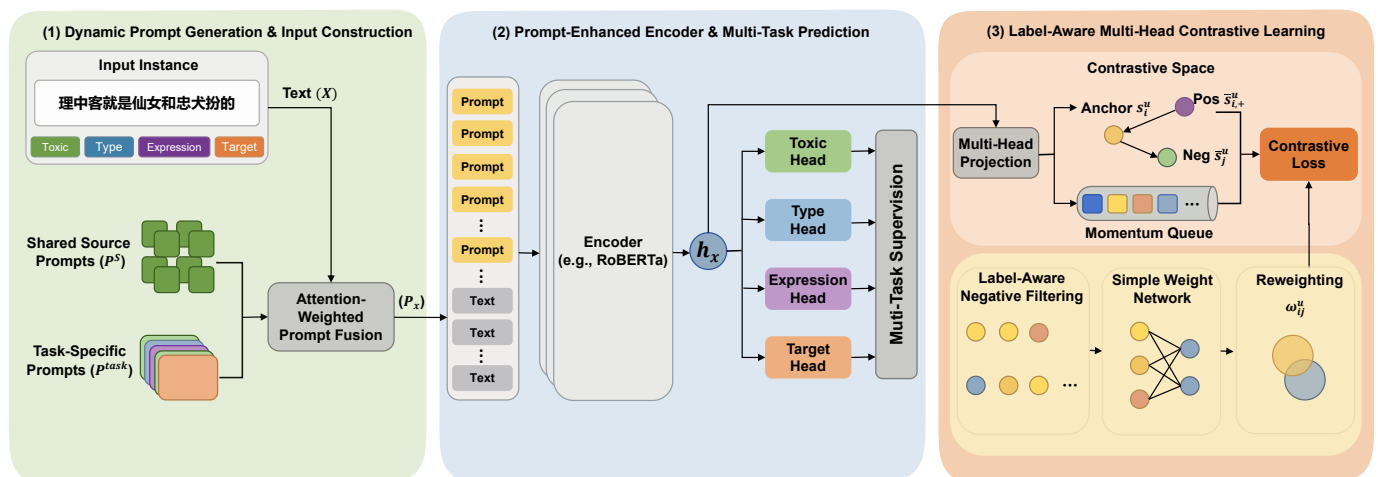


Figure 2. Overall architecture of PEMTCL, comprising three modules: (1) dynamic prompt generation and input construction, (2) prompt-enhanced encoding and multi-task prediction, and (3) label-aware multi-head contrastive learning. The Chinese input example means “Those neutral commentators are just women using beauty and men being submissive in disguise.” Arrows indicate the flow of representations and supervision, and colors distinguish the principal modules, task-specific components, and contrastive samples.

4.2. Design Rationale and Methodological Distinction

Fine-grained toxic language detection involves heterogeneous but correlated supervision: toxicity identification is binary, toxic type and expression type use distinct label spaces, and targeted group detection is multi-label. This structure creates three coupled requirements—sample-specific input adaptation, joint learning across subtasks, and task-aware representation regularization. PEMTCL addresses these requirements in a unified pipeline. Dynamic prompts condition the shared encoder on the current input, multi-task prediction organizes the heterogeneous supervision, and label-aware contrastive learning regularizes the resulting representations according to task-specific label relations. The resulting framework is therefore distinguished by the interaction among its components rather than by an independent use of prompt tuning, multi-task learning, or contrastive learning.

Table 1 summarizes the main methodological differences from representative related approaches. Existing methods generally address only one part of the problem, whereas PEMTCL jointly supports input-conditioned prompting, four-subtask prediction, task-specific contrastive spaces, task-specific label filtering, and class-aware negative weighting. This task-structured integration is adopted because a disconnected mechanism would address only an isolated source of error while leaving the remaining supervision and representation mismatches unresolved.

The dynamic prompt mechanism provides sample-specific adaptation through $P_x = \sum_{j=1}^{M+1} \alpha_j(x) C_j$. Because $\alpha_j(x)$ depends on the current text and the aggregated task prior, the encoder receives different prompt combinations for different samples;

a fixed prompt is recovered when the coefficients are constant. During optimization, the contribution of each candidate prompt is modulated by its input-dependent weight, directing more learning capacity toward prompts that are relevant to the current sample and reducing interference from less relevant task cues.

Table 1. Comparison of representative methods. ✓ denotes full support, ◦ denotes partial support, and “–” indicates that the corresponding mechanism is not included.

Method	Dynamic Prompt	Joint Tasks	Multi-Head CL	Label Filtering	Class Weighting
TKE [5]	–	–	–	–	–
SoftPrompt [10]	◦	–	–	–	–
MTL [6]	–	✓	–	–	–
LAHN [7]	–	–	–	◦	–
PEMTCL	✓	✓	✓	✓	✓

The label-aware contrastive module modifies the representation geometry at the output side. Conventional contrastive learning repels every retained negative, which can be harmful when two samples share labels in some subtasks. Task-specific label filtering removes such unreliable negative relations, while ω_{ij}^k scales the remaining terms according to similarity, label overlap, and class distribution. The task-specific projection heads further separate heterogeneous supervision into distinct contrastive spaces. Together, these operations suppress contradictory repulsive constraints and concentrate optimization on genuinely confusable negatives, providing a direct mechanism for improving fine-grained representation discrimination.

4.3. Dynamic Prompt Generation

Different subtasks focus on different semantic cues. If the raw text is directly fed into the shared encoder, the model must simultaneously accommodate the discriminative requirements of different tasks within the same representation space. To this end, PEMTCL introduces a dynamic prompt generation mechanism at the input stage, which constructs task-relevant soft prompts based on the current sample and incorporates them as part of the encoder input.

The model first obtains the text representation through the embedding layer:

$$E_x = \text{Embedding}(x) \in \mathbb{R}^{n \times d}, \quad (1)$$

where n is the text length and d is the hidden dimension. The model maintains two sets of learnable prompt parameters. The first is shared source prompts: $P^S = \{p_1^S, \dots, p_M^S\}$. The second is task-specific prompts: $P^{\text{task}} = \{p_{\text{tox}}, p_{\text{type}}, p_{\text{expr}}, p_{\text{tgt}}\}$. Here, M denotes the number of shared source prompts, and both source and task-specific prompts have dimensions $r \times d$, where r is the prompt length. The shared source prompts capture cross-task general toxic semantic cues, while the task-specific prompts correspond to the discriminative requirements of individual subtasks. It is worth noting that the task-specific prompts are learnable continuous vectors rather than ground-truth labels of samples. The ground-truth labels are used only during the training phase for computing supervised losses, while the model receives only the input text during inference.

PEMTCL first generates a lightweight and interpretable global task prior. This prior serves a dual purpose: as a candidate prompt on one hand, and as an attention conditioning factor on the other, enabling the model to dynamically adapt to the specific requirements of

different samples while maintaining task generality. To this end, PEMTCL aggregates the four task-specific prompts by averaging to obtain P^G :

$$P^G = \frac{1}{4}(p_{\text{tox}} + p_{\text{type}} + p_{\text{expr}} + p_{\text{tgt}}). \quad (2)$$

To construct a prompt pool containing both general semantic priors and task-level priors, enabling the model to dynamically fuse the most suitable instance-level prompts for each sample and thereby improve adaptability and performance under multi-task scenarios, we define the candidate prompt set $\mathcal{C} = \{C_1, \dots, C_{M+1}\}$, where $C_j = p_j^S$ for $1 \leq j \leq M$, and $C_{M+1} = P^G$. The model then applies pooling to the text representation, candidate prompts, and global task prompt, respectively, obtaining the text query vector $q_x = \text{Pooling}(E_x)$, candidate prompt vectors $c_j = \text{Pooling}(C_j)$, and the task prompt vector $g = \text{Pooling}(P^G)$, where $q_x, c_j, g \in \mathbb{R}^d$. On this basis, the relevance score a_j between the j -th candidate prompt and the current input instance is computed as:

$$a_j = v^\top \tanh(W_q q_x + W_c c_j + W_g g + b), \quad (3)$$

and then normalized via softmax:

$$\alpha_j = \frac{\exp(a_j)}{\sum_{l=1}^{M+1} \exp(a_l)}, \quad (4)$$

where $W_q, W_c, W_g \in \mathbb{R}^{d_a \times d}$, $v, b \in \mathbb{R}^{d_a}$, and d_a is the attention hidden dimension. α_j represents the matching weight between the current input and the j -th candidate prompt, allowing different samples to obtain differentiated weight distributions. P^G plays a dual role in this process: as one of the candidate prompts, it directly participates in the generation of instance-level prompts; simultaneously, as an attention condition, it modulates the matching relationship between shared source prompts and the current task set. Since the weights are dynamically computed based on the current input, the model is capable of generating adaptive prompts for different samples.

Based on the above weights, the model performs a weighted summation over the candidate prompts to obtain the instance-level prompt for the current sample:

$$P_x = \sum_{j=1}^{M+1} \alpha_j C_j, \quad (5)$$

where $P_x \in \mathbb{R}^{r \times d}$ represents the dynamic selection of candidate prompts conditioned on the current input. Finally, the model concatenates the instance-level prompt P_x before the original text embedding E_x to form the prompt-enhanced input representation $\tilde{E}_x = [P_x; E_x]$, where $\tilde{E}_x \in \mathbb{R}^{(r+n) \times d}$ retains the original text content while incorporating continuous prompt information related to the current sample and task set at the input stage. Subsequently, $\tilde{E}_x$ is fed into the shared pre-trained language model encoder for downstream multi-task contextual modeling.

4.4. Multi-Task Learning Framework

To fully model the fine-grained semantic information in toxic text detection, we design a multi-task learning module on top of the prompt-enhanced representations. Unlike conventional methods that merely determine whether a text is toxic, we decompose the toxic language detection task into multiple interrelated subtasks. Through joint modeling with a shared encoder and task-specific prediction heads, the model can simultaneously learn

coarse-grained and fine-grained toxicity features within a unified semantic representation space, thereby improving its recognition capability for complex toxic expressions.

Specifically, the prompt-enhanced input $\tilde{E}_x$ is first passed through the pre-trained language model encoder to obtain contextual representations, from which a sentence-level semantic representation is extracted as the shared feature for multi-task learning. On this basis, we construct task-specific prediction heads for the four subtasks of toxicity identification, toxic type discrimination, expression type detection, and targeted group detection, enabling the model to retain the differentiated discriminative capability of each task while sharing cross-task common semantics. During training, the model adopts a joint optimization strategy that unifies the losses of individual subtasks into a multi-task learning objective, selecting the appropriate classification loss function according to the label format of each subtask.

4.5. Label-Aware Multi-Head Contrastive Learning

Conventional contrastive learning constructs positive and negative samples based on category labels. However, samples in toxic text detection often possess multi-dimensional label relationships, and simply treating samples with not entirely matching labels as negatives can easily introduce false negative samples, causing semantically related samples to be erroneously separated. Therefore, building upon multi-task learning, we propose a label-aware multi-head contrastive learning module that employs multi-head projection, momentum queue construction, negative sample filtering, and sample reweighting mechanisms to achieve more fine-grained contrastive constraints.

The multi-head projection method maps the shared semantic representation into multiple task-specific contrastive spaces, where each projection head models differentiated semantic features for the respective tasks of toxicity identification, toxic type discrimination, expression type detection, and targeted group detection. In contrastive learning, the current sample serves as the anchor, and its momentum-encoded representation is directly used as the positive sample, thereby avoiding label noise. Label information is used solely for negative sample filtering and reweighting, selecting samples with significant semantic differences and minimal label overlap as negatives to construct fine-grained constraints. For the k -th task, the contrastive representation is defined as:

$$u_i^k = \text{Norm}(g_k(h_i)), \quad k \in \{\text{tox}, \text{type}, \text{expr}, \text{tgt}\}, \quad (6)$$

where $g_k(\cdot)$ denotes the projection head for the k -th task, and $\text{Norm}(\cdot)$ denotes vector normalization.

Momentum Queue Construction: Due to the limited training batch size, constructing contrastive samples solely from the current batch can easily lead to insufficient candidate samples, particularly when the class distribution is imbalanced or fine-grained samples are scarce, which adversely affects contrastive learning performance. To this end, we maintain a momentum queue for each task to cache historical sample representations along with their labels, serving as an extended contrastive sample set for the current batch. This mechanism enriches negative sample sources without increasing the batch size, improving training stability and representation discriminability. The momentum encoder parameters are updated via exponential moving average (EMA):

$$\theta_m \leftarrow \mu \theta_m + (1 - \mu) \theta, \quad (7)$$

where θ_m and θ denote the parameters of the momentum encoder and the current encoder, respectively, and μ is the momentum coefficient.

Label-Aware Negative Sample Filtering: In multi-label, multi-task scenarios, candidate negative samples may share partial semantic associations with the anchor sample. If directly used as negatives in contrastive learning, they can easily introduce false negatives or noise and disrupt the structure of the representation space. To address this issue, we design a label-aware negative sample filtering mechanism that removes unreliable negative samples based on the matching relationships among multi-task labels. Specifically, for the toxicity identification, toxic type discrimination, and expression type detection tasks, only samples with different labels are selected as negatives, provided that the labels are valid. For the targeted group detection task, since the labels are multi-hot vectors, the model evaluates whether candidate negative samples overlap with the anchor sample on certain key label dimensions, and reduces or removes the negative constraints of such samples in contrastive learning. The label overlap rate is defined as:

$$\rho_{ij}^{\text{tgt}} = \frac{|y_i^{\text{tgt}} \cap y_j^{\text{tgt}}|}{|y_i^{\text{tgt}} \cup y_j^{\text{tgt}}|}, \quad (8)$$

where y_i^{tgt} and y_j^{tgt} denote the multi-valued labels for targeted group detection. When ρ_{ij}^{tgt} exceeds the threshold δ , sample j shares substantial target overlap with the anchor sample and is excluded as a strong negative from the contrastive loss of the current anchor; otherwise, sample j is retained as a candidate negative sample.

Sample Reweighting Mechanism: To fully exploit the discriminative value of different negative samples, PEMTCL further introduces a sample reweighting mechanism after negative sample filtering. This mechanism assigns adaptive weights to candidate negative samples through a lightweight weighting network, strengthening the contrastive constraints on hard negative samples while reducing the interference from noisy or semantically overlapping samples.

Let q_j^k denote the representation of queue sample j in the k -th task and u_i^k denote the anchor sample representation. Their similarity is computed as:

$$s_{ij}^k = (u_i^k)^\top q_j^k. \quad (9)$$

To mitigate the impact of class distribution imbalance, we further compute a class distribution factor for each queue sample:

$$d_j^k = \frac{1}{\max(1, \|y_j^k\|_1)} \sum_{c=1}^{C_k} \frac{y_{j,c}^k}{\sqrt{f_c^k + \epsilon}}, \quad (10)$$

where C_k denotes the number of classes for the k -th task, f_c^k denotes the global frequency of the c -th class label for the k -th task in the training set, and ϵ is a smoothing term. For single-valued labels, this term represents the frequency weight of the active class; for multi-valued labels, it represents the average frequency weight across all active classes.

Subsequently, the model takes the sample similarity, label overlap information, and class distribution factor as inputs to the weighting network, and generates normalized weight scores for candidate negative samples through a lightweight feedforward network:

$$a_{ij}^k = w_2^\top \text{ReLU}(W_1[s_{ij}^k; \rho_{ij}^k; d_j^k] + b_1) + b_2, \quad (11)$$

$$\omega_{ij}^k = \frac{\exp(a_{ij}^k)}{\sum_{l \in \mathcal{N}_i^k} \exp(a_{il}^k)}, \quad (12)$$

where ω_{ij}^k is the normalized weight, and $\mathcal{N}_i^k$ denotes the candidate negative sample set for sample i in the k -th task.

4.6. Training Objective

The overall loss of PEMTCL consists of a weighted combination of the multi-task supervised loss $\mathcal{L}_{\text{mtl}}$ and the label-aware contrastive loss $\mathcal{L}_{\text{con}}$:

$$\mathcal{L} = \mathcal{L}_{\text{mtl}} + \lambda \mathcal{L}_{\text{con}}, \quad (13)$$

where λ is a hyperparameter.

The multi-task supervised loss jointly optimizes four related tasks through hierarchical constraints and homoscedastic uncertainty weighting, balancing the differences in label space, sample size, and learning difficulty across tasks:

$$\mathcal{L}_{\text{mtl}} = \sum_{k \in \mathcal{K}} (\exp(-\eta_k) \mathcal{L}_k + \eta_k), \quad (14)$$

$$\mathcal{L}_k = \frac{1}{Z_k} \sum_{i=1}^N m_i^k \ell_k(z_i^k, y_i^k), \quad Z_k = \sum_{i=1}^N m_i^k, \quad (15)$$

where η_k is a learnable parameter that adjusts the relative weight of the k -th task loss in joint optimization. This formulation equivalently represents the original homoscedastic uncertainty weighting objective using log-variance parameters. $\mathcal{L}_k$ denotes the classification loss for the k -th task, N is the batch size, $m_i^k \in \{0, 1\}$ is the valid label indicator, and $\ell_k(\cdot)$ denotes the classification loss function for the k -th task. For $k = \text{tox}$, $\ell_k(\cdot)$ is the cross-entropy loss; for $k \in \{\text{type}, \text{expr}, \text{tgt}\}$, $\ell_k(\cdot)$ is the binary cross-entropy loss with logits. Z_k normalizes over the number of valid samples. If no valid samples exist for a particular task in the current batch, the corresponding task loss does not participate in the update for that batch.

The label-aware contrastive loss constructs a representation space with intra-class compactness, inter-class separability, and high sensitivity to confusable samples through task-specific contrastive learning, thereby enhancing the robustness and generalization of fine-grained toxic language detection:

$$\mathcal{L}_{\text{con}} = \sum_{k \in \mathcal{K}} \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathcal{L}_{\text{con}}^k(i), \quad (16)$$

where $\mathcal{V}_k = \{i \mid m_i^k = 1\}$ denotes the set of samples with valid labels that can serve as anchors for the k -th task. $\mathcal{L}_{\text{con}}^k(i)$ is the contrastive loss for sample i under task k , computed as:

$$\mathcal{L}_{\text{con}}^k(i) = -\log \frac{\mathcal{S}_{\text{pos}}^k(i)}{\mathcal{S}_{\text{pos}}^k(i) + \mathcal{S}_{\text{neg}}^k(i)}, \quad (17)$$

where $\mathcal{S}_{\text{pos}}^k(i)$ denotes the positive contribution for sample i under task k in contrastive learning, and $\mathcal{S}_{\text{neg}}^k(i)$ denotes the weighted negative contribution for sample i in contrastive learning.

5. Experiments

5.1. Datasets

To evaluate the effectiveness of PEMTCL on fine-grained toxic language detection, we conduct experiments on two publicly Chinese datasets, ToxiCN [5] and COLD [21].

ToxiCN is a fine-grained Chinese toxic language dataset covering four subtasks: toxicity identification, toxic type discrimination, expression type detection, and targeted group detection. We directly adopt its original data splits and label scheme.

The COLD dataset provides only binary toxicity labels, which are insufficient to support fine-grained task modeling for toxic type, expression type, and target group dimensions. Therefore, we retain the original toxicity labels and further annotate three additional subtask labels for toxic samples, while uniformly assigning a label of None to non-toxic samples in the corresponding dimensions, constructing the fine-grained extended dataset FG-COLD. Specifically, we first establish annotation guidelines covering label definitions, representative examples, and boundary cases to reduce ambiguity in fine-grained category judgment. Dual independent annotation combined with expert adjudication is then employed to determine the final labels. Annotation quality is ensured through pre-annotation, random re-examination, and anomaly sample inspection. Annotation consistency is measured using the Kappa coefficient, reaching 0.72, 0.76, and 0.83 for toxic type discrimination, expression type detection, and targeted group detection, respectively, indicating that FG-COLD possesses satisfactory annotation consistency and reliability.

The statistics of the ToxiCN and FG-COLD datasets are presented in Table 2. Since targeted group detection is a multi-label task, the sum of samples across target groups may exceed the total number of hate speech samples.

Table 2. Statistics of ToxiCN and FG-COLD.

Task Type	Label	ToxiCN	FG-COLD
Toxic	Non-Toxic	5550	19,439
	Toxic	6461	18,041
Toxic Type (among toxic)	General Offensive Language	816	2421
	Hate Speech	5645	15,620
Expression (among hate)	Explicitness	2737	3019
	Implicitness	1995	11,418
	Reporting	913	1183
Target (among hate)	Region	1290	4493
	Gender	2302	5697
	Race	1877	7221
	LGBTQ	1075	947
Overview	Total Samples	12,011	37,480

The class distributions in both datasets are inherited from the source corpora rather than created through class-balanced resampling. In the adopted taxonomy, General Offensive Language covers toxic content that is not directed at a protected group, whereas Reporting is assigned when offensive content is quoted, described, or discussed without direct endorsement. These narrower annotation conditions contribute to their smaller sample sizes. The natural class distributions are retained to preserve the characteristics of the source corpora, while class-aware negative weighting is used to reduce the dominance of frequent classes during contrastive learning.

FG-COLD contains more samples overall but fewer LGBTQ instances (947) than ToxiCN (1075). The two datasets were collected independently, and FG-COLD was obtained by adding fine-grained annotations to the original COLD corpus without category-level resampling. Thus, subgroup counts reflect the composition of the underlying corpora and do not necessarily scale with the total dataset size.

5.2. Experimental Setup

Implementation Details: All experiments are implemented in PyTorch 1.7.0 and conducted on an NVIDIA GeForce RTX 3090 GPU. The backbone is Chinese-RoBERTa-wwm-ext, with a maximum input length of 128 and a batch size of 32. The AdamW optimizer is used with a learning rate of 2×10^{-5} and a weight decay of 0.01. To mitigate the impact of random initialization, all experiments are repeated under five different random seeds (41–45), and the average results are reported. For the dynamic prompt generation module, the prompt length r is set to 16, the number of shared source prompts M is set to 4, and the attention hidden dimension d_a is set to 256. For label-aware contrastive learning, the temperature coefficient τ is set to 0.07, the momentum coefficient μ is set to 0.999, the momentum queue size is set to 1024, and a 1-epoch queue warm-up is performed at the beginning of training. The target label overlap threshold δ is set to 0.4, and the contrastive loss weight λ is set to 0.1.

Hyperparameters were selected through a staged search on the validation sets. Prompt-related parameters were examined first, followed by contrastive-learning parameters, with one parameter varied at a time and the remaining parameters fixed at their current best values. The candidate values were $\{1 \times 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}\}$ for the learning rate; $\{4, 8, 16, 32\}$ for the prompt length r ; $\{2, 4, 6, 8\}$ for the number of prompts M ; $\{64, 128, 256, 512\}$ for the attention dimension d_a ; $\{0.03, 0.05, 0.07, 0.10, 0.20\}$ for the temperature τ ; $\{256, 512, 1024, 2048\}$ for the queue size; $\{0.2, 0.3, 0.4, 0.5, 0.6\}$ for the overlap threshold δ ; and $\{0.05, 0.10, 0.20, 0.30, 0.50\}$ for the contrastive weight λ . For ToxiCN, the validation criterion was the average Weighted-F1 across the four subtasks, whereas for FG-COLD it was the average Macro-F1 across the four subtasks. The overall selection score was the mean of the two dataset-level scores. The selected configuration achieved validation scores of 74.84 on ToxiCN, 65.30 on FG-COLD, and 70.07 overall. The test sets were used only for the final evaluation.

Evaluation Metrics: To ensure comparability with prior work, we follow the evaluation protocols of the original ToxiCN and COLD papers. Specifically, Weighted-F1 is adopted for ToxiCN, and Macro-F1 is adopted for FG-COLD. Statistical significance is assessed using paired t -tests, comparing PEMTCL with the closest non-PEMTCL baseline under the same backbone on each subtask. * denotes $p < 0.01$ and † denotes $p < 0.05$.

5.3. Baseline Comparisons

We select four categories of methods as baselines. For LLM-based methods, we directly cite the results reported in their original papers; for all other baselines, we reproduce the results following the original experimental configurations. Recent advances in large language models have further expanded the toxic language detection toolbox through chain-of-thought reasoning [22], persona-based social reasoning [23], model behavioral analysis [24], and confidence calibration under annotation disagreement [25].

- **LLM-based methods:** GPT-Zero, GPT-CoT, GPT-Unic [26], and RCT [27].
- **Prompt-based methods:** Prompt-BERT and Prompt-RoBERTa [28].
- **Fine-grained methods:** TKE-BERT, TKE-RoBERTa [5], ToXCL-BART, and ToXCL-T5 [16].
- **Multi-task multi-label methods:** MTL-BERT, MTL-RoBERTa [6], NLPineers-RoBERTa, and NLPineers-BERT [29].

The performance comparison on ToxiCN and FG-COLD is presented in Tables 3 and 4. “–” indicates that the baseline does not support the corresponding task under its original setting. The best results are highlighted in **bold** and the second best are underlined. The experimental results on ToxiCN and FG-COLD yield the following observations.

First, on ToxiCN, PEMTCL achieves the best performance on three fine-grained sub-tasks, i.e., toxic type discrimination, expression type detection, and targeted group detection, demonstrating that the PEMTCL framework can more effectively model the subtle semantic differences in toxic text. Although PEMTCL is slightly below RCT on the binary toxicity identification task, RCT relies on external large model knowledge, whereas our method requires no external knowledge sources, achieving a favorable trade-off between practicality and performance. This makes PEMTCL more suitable for real-world scenarios with resource constraints or high inference efficiency requirements.

Second, on FG-COLD, PEMTCL achieves the best performance across all four toxic language detection tasks, with particularly notable improvements on the three fine-grained subtasks. Considering that the fine-grained labels in FG-COLD are re-annotated on top of the original COLD toxicity labels, resulting in more complex task distributions and class boundaries, the consistent improvements in PEMTCL further validate the effectiveness of the proposed method. It is worth noting that the GPT-based and RCT methods do not report results on this dataset and are therefore excluded from the table.

Table 3. Results on ToxiCN (Weighted-F1, %). Bold indicates the best result, and underlining indicates the second-best result. * denotes $p < 0.01$, and † denotes $p < 0.05$, compared with the closest non-PEMTCL baseline under the same backbone.

Model	Toxic	Toxic Type	Expression	Target
GPT-Zero	74.3	–	–	–
GPT-CoT	75.1	–	–	–
GPT-Unic	65.8	–	–	–
RCT	83.2	<u>79.0</u>	<u>60.1</u>	<u>76.6</u>
Prompt-BERT	74.8	66.2	30.3	–
Prompt-RoBERTa	74.9	65.8	31.9	–
TKE-BERT	79.5	75.1	52.1	72.9
TKE-RoBERTa	80.8	77.0	54.3	73.9
ToXCL-BART	77.6	66.3	24.4	–
ToXCL-T5	78.6	73.1	37.6	–
MTL-BERT	80.2	75.6	52.3	74.0
MTL-RoBERTa	80.9	76.6	53.6	75.0
NLPineers-BERT	80.4	76.0	56.9	75.8
NLPineers-RoBERTa	81.4	78.5	57.0	75.4
PEMTCL	<u>83.1</u>	79.4 †	60.6 *	77.8 *

Table 4. Results on FG-COLD (Macro-F1, %). Bold indicates the best result, and underlining indicates the second-best result. * denotes $p < 0.01$, and † denotes $p < 0.05$, compared with the closest non-PEMTCL baseline under the same backbone.

Model	Toxic	Toxic Type	Expression	Target
Prompt-BERT	78.1	43.9	26.6	–
Prompt-RoBERTa	79.7	44.8	23.5	–
TKE-BERT	81.0	57.0	48.3	61.3
TKE-RoBERTa	81.8	57.1	50.8	61.1
ToXCL-BART	81.3	53.2	35.1	–
ToXCL-T5	81.7	52.5	35.3	–
MTL-BERT	80.8	55.4	46.8	62.0
MTL-RoBERTa	81.2	56.3	48.1	63.1
NLPineers-BERT	<u>82.0</u>	58.8	49.6	64.3
NLPineers-RoBERTa	<u>82.0</u>	<u>59.4</u>	<u>51.2</u>	<u>65.8</u>
PEMTCL	82.4 *	61.3 *	51.9 *	67.1 †

Third, taking the experimental results on both datasets together, PEMTCL demonstrates favorable stability across different data distributions and evaluation metrics, with particularly significant improvements on fine-grained subtasks such as toxic type discrimination, expression type detection, and targeted group detection. These results indicate that the proposed method can effectively model semantic correlations and complementary relationships among multiple tasks, and further optimizes task-specific representation spaces through label-aware contrastive learning, thereby enhancing the model's discriminability for fine-grained semantic distinctions in toxic language.

5.4. Ablation Study

To evaluate the contribution of each module, we conduct ablation experiments on ToxiCN and FG-COLD. The results are presented in Tables 5 and 6. The following variants are designed:

- **w/o Dynamic Prompt** removes the dynamic prompt generation mechanism and directly feeds the raw text embeddings into the shared encoder.
- **w/o Contrastive** removes the entire label-aware multi-head contrastive learning module, retaining only the multi-task supervised loss.
- **w/o Negative Filtering** retains the contrastive learning module but removes the label-aware negative sample filtering mechanism, treating all momentum queue samples as candidate negatives.
- **w/o Hard Negative Weighting** retains contrastive learning and negative sample filtering but removes the hard negative weighting network, applying equal weights to all candidate negative samples.

Table 5. Ablation results on ToxiCN (Weighted-F1, %). The best result in each column is shown in bold.

Model Variant	Toxic	Toxic Type	Expression	Target
PEMTCL (Full)	83.1	79.4	60.6	77.8
w/o Dynamic Prompt	82.4	78.4	58.8	76.6
w/o Contrastive	82.3	78.8	59.1	76.9
w/o Negative Filtering	82.7	78.9	59.6	76.5
w/o Hard Negative Weighting	82.6	79.0	59.5	77.1

Table 6. Ablation results on FG-COLD (Macro-F1, %). The best result in each column is shown in bold.

Model Variant	Toxic	Toxic Type	Expression	Target
PEMTCL (Full)	82.4	61.3	51.9	67.1
w/o Dynamic Prompt	81.7	60.0	50.3	66.3
w/o Contrastive	81.5	59.3	50.1	66.4
w/o Negative Filtering	82.0	60.3	51.2	65.7
w/o Hard Negative Weighting	81.8	60.4	50.7	66.6

The results show that the full model achieves the best performance across all subtasks on both datasets, and removing any module leads to performance degradation. From the perspective of major modules, removing either the dynamic prompt generation or the contrastive learning module results in notable performance drops, with larger decreases on fine-grained subtasks than on the binary toxicity identification task. This effect is particularly pronounced on FG-COLD, which has a more complex label distribution, indicating that both modules make significant and complementary contributions to the framework.

The negative sample filtering and sample reweighting mechanisms have relatively smaller impacts on model performance.

5.5. Effect of Training Data Scale

To evaluate the impact of training data scale on model performance, we select TKE-RoBERTa and MTL-RoBERTa, which achieve competitive performance on ToxiCN, and train them using 10%, 25%, 50%, 75%, and 100% of the original training data, respectively. The results are shown in Figure 3.

The results demonstrate that as the proportion of training data increases, the performance of all models improves steadily across the four subtasks. PEMTCL achieves the best results at all data proportions and on all subtasks, indicating its favorable stability under varying data scales. Particularly on fine-grained subtasks such as toxic type discrimination, expression type detection, and targeted group detection, the advantage of PEMTCL is more pronounced, suggesting that the dynamic prompt mechanism and label-aware contrastive learning can enhance task-relevant semantic modeling and representation discriminability even with limited labeled samples, thereby improving model robustness under low-resource training conditions.

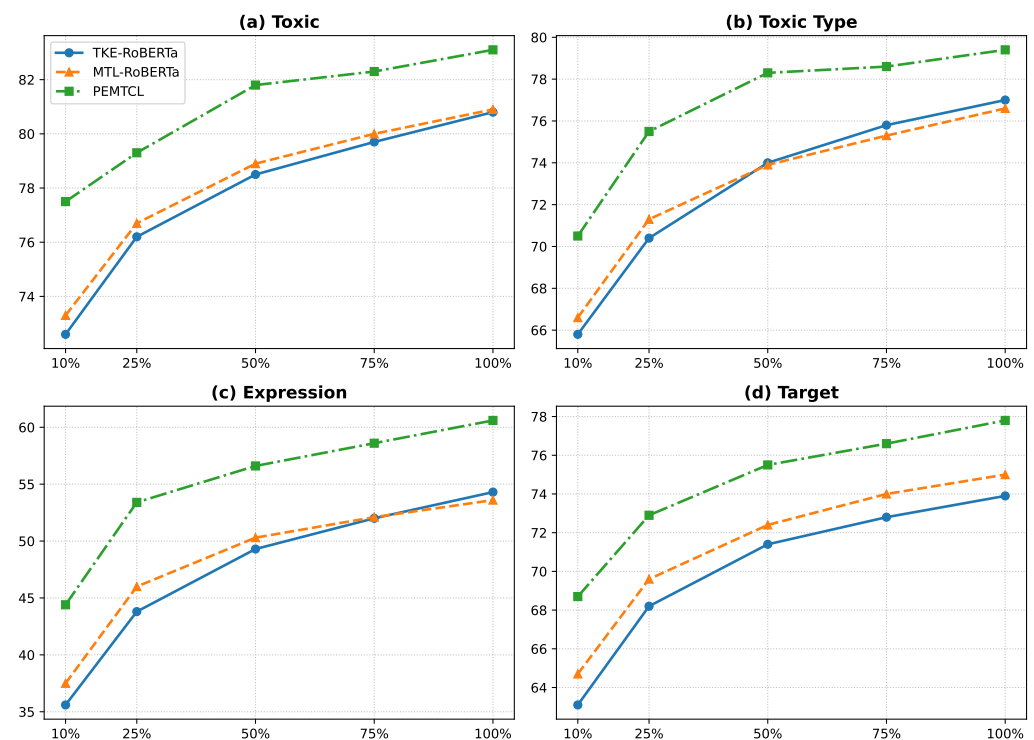


Figure 3. Effect of training data scale on model performance.

5.6. Case Study

We select four representative samples from the ToxiCN dataset for case study analysis. The results are presented in Table 7. The original texts are in Chinese; English translations are provided for readability.

Case 1 contains implicit gender discrimination. MTL misclassifies it as Non-Toxic, while TKE correctly identifies toxicity but misclassifies the expression type as Explicitness. PEMTCL correctly identifies both the Implicitness expression and the Gender target, demonstrating stronger semantic comprehension of implicit discriminatory language.

Table 7. Case study on ToxiCN. The texts in the Text column are English translations of the original Chinese samples. Predictions follow the format: Toxic/Toxic Type/Expression/Target. “–” indicates that the label is not applicable. ✓ denotes a fully correct prediction and × denotes an incorrect prediction.

Case	Text	Ground Truth	MTL-RoBERTa	TKE-RoBERTa	PEMTCL
1	What’s the issue? Hire a man and a woman—she drinks tea while he does twice the work.	Toxic/Hate/ Implicit/Gender	Non-Toxic ×	Toxic/Hate/ Explicit/Gender ×	Toxic/Hate/ Implicit/Gender ✓
2	Just enact a new law: any woman who marries a foreigner automatically loses Chinese citizenship.	Toxic/Hate/Explicit/ Gender + Race	Toxic/Hate/Explicit/ Race ×	Toxic/Hate/Explicit/ Race ×	Toxic/Hate/Explicit/ Gender + Race ✓
3	Have some dignity, use your brain.	Toxic/Offensive/–/–	Toxic/Hate/–/– ×	Toxic/Hate/Explicit/ Others ×	Toxic/Offensive/–/– ✓
4	Can you distinguish regional discrimination from stereotypes?	Non-Toxic/–/–/–	Toxic ×	Toxic/Hate/ Reporting/Region ×	Non-Toxic/–/–/– ✓

Case 2 simultaneously involves gender-based and race-based discrimination. Both MTL and TKE identify only the Race target, missing the Gender target. PEMTCL accurately predicts the composite target Gender + Race, validating that the label-aware multi-head contrastive learning helps capture complex multi-label semantics.

Case 3 is General Offensive Language rather than Hate Speech. Both MTL and TKE misclassify it as Hate Speech, and TKE additionally assigns incorrect fine-grained attributes such as Explicitness and Others. PEMTCL correctly identifies it as General Offensive Language without adding excessive specific categories, demonstrating its accurate judgment at class boundaries.

Case 4 is actually a non-toxic text that constitutes a clarification of concepts. MTL misclassifies it as toxic, and TKE not only misclassifies it as toxic but also erroneously assigns fine-grained attributes such as Reporting and Region. PEMTCL correctly identifies it as Non-Toxic, demonstrating its ability to effectively distinguish between “mentioning discriminatory concepts” and “actually expressing discriminatory speech,” reflecting stronger contextual comprehension.

In summary, PEMTCL outperforms the baseline models in implicit expression recognition, composite target identification, class boundary discrimination, and negation context comprehension, fully demonstrating the superiority of the multi-task contrastive learning framework for fine-grained toxic language detection.

6. Limitations

The current evaluation is limited to ToxiCN and FG-COLD, both of which are Chinese-language datasets, and all experiments use Chinese-RoBERTa-wwm-ext as the shared encoder. The reported results therefore do not establish cross-lingual, cross-domain, cross-platform, or cross-encoder generalization. The natural data distributions also contain relatively few instances in categories such as Reporting and LGBTQ, which may increase estimation variance and restrict the coverage of less frequent linguistic phenomena and demographic groups. Class-aware negative weighting reduces the influence of class imbalance during training but does not eliminate the need for broader and more balanced evaluation data.

The overlap threshold δ is fixed at 0.4 after validation-based selection. Across the evaluated values from 0.2 to 0.6, the overall validation score ranged from 69.54 to 70.07, with 0.4 achieving the best result. However, the relatively small differences among neighboring values indicate that the most suitable boundary may depend on the dataset, class distribution, sample difficulty, or training stage. Fine-grained annotation also involves subjective distinctions, particularly for implicit expressions, reporting cases, and texts involving multiple target groups, despite the use of independent annotation, expert adjudication, and consistency checks. In addition, dynamic prompt generation and the momentum queue introduce additional memory and computational costs. The LLM-based baseline results are taken from their original studies rather than reproduced under a fully unified execution environment, which limits strict experimental comparability.

7. Future Work

Future work will evaluate PEMTCL across languages, domains, platforms, and encoder families [2,30–34]. We will investigate sample-adaptive or learnable overlap thresholds, improved learning strategies for rare classes, and demographic fairness [35,36], together with more consistent reproduction of LLM-based baselines. Further work will also consider more efficient prompt and queue designs, multimodal and context-enhanced detection [37], and robustness to character substitution, homophonic perturbation, and other evasion strategies [38–41].

8. Conclusions

We proposed PEMTCL, a prompt-enhanced multi-task contrastive learning framework for fine-grained toxic language detection. By integrating dynamic prompt generation, multi-task joint optimization of four subtasks, and label-aware multi-head contrastive learning with negative sample filtering and hard negative reweighting, the framework achieves the best performance on most subtasks across ToxiCN and FG-COLD, with particularly notable improvements in fine-grained dimensions.

The ablation, training-scale, and case study results further indicate that dynamic prompt generation and label-aware contrastive learning provide complementary benefits for sample-specific adaptation and fine-grained representation discrimination.

Author Contributions: Conceptualization, S.J. and X.F.; methodology, S.J.; software, S.J.; validation, Z.H. and P.C.; formal analysis, S.J.; investigation, Z.H. and P.C.; resources, X.F.; data curation, S.J. and Z.H.; writing—original draft preparation, S.J.; writing—review and editing, X.F., Z.H. and P.C.; visualization, P.C.; supervision, X.F.; project administration, X.F.; funding acquisition, X.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Education Humanities and Social Sciences General Project (24XIJAZH001); Xinjiang Normal University Smart Education Engineering Center (XJNU-ZHJY202402); the Major Science and Technology Special Project of the Xinjiang Uygur Autonomous Region (22023A03001-2); Research on Hyperspectral Image Classification Methods Based on Spatial Structure (2022D01A229); and the Doctoral Research Start-up Fund of Xinjiang Normal University (XJNUBS202407).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: ToxiCN is publicly available from its original publication. The re-annotated FG-COLD dataset and source code are available at <https://github.com/jin-shan/PEMTCL> (accessed on 5 August 2026). The complete FG-COLD annotation guideline document is included in the same source-code repository.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tonneau, M.; Liu, D.; Malhotra, N.; Hale, S.A.; Fraiberger, S.; Orozco-Olvera, V.; Röttger, P. HateDay: Insights from a Global Hate Speech Dataset Representative of a Day on Twitter. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 2297–2321.
2. Muhammad, S.H.; Abdulmumin, I.; Ayele, A.A.; Adelani, D.I.; Ahmad, I.S.; Aliyu, S.M.; Röttger, P.; Oppong, A.; Bukula, A.; Chukwuneke, C.I.; et al. AfriHate: A Multilingual Collection of Hate Speech and Abusive Language Datasets for African Languages. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 1854–1871.
3. Zeng, J.; Yang, L.; Wang, Z.; Sun, Y.; Lin, H. Sheep’s Skin, Wolf’s Deeds: Are LLMs Ready for Metaphorical Implicit Hate Speech? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 16657–16677.
4. Jin, Y.; Wanner, L.; Koya, A.M. What the #?!: Disentangling Hate Across Target Identities. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 199–221.
5. Lu, J.; Xu, B.; Zhang, X.; Min, C.; Yang, L.; Lin, H. Facilitating Fine-grained Detection of Chinese Toxic Language: Hierarchical Taxonomy, Resources, and Benchmarks. In *Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2023; pp. 16235–16250.
6. Pan, R.; García-Díaz, J.A.; Valencia-García, R. Spanish MTLHateCorpus 2023: Multi-task learning for hate speech detection to identify speech type, target, target group and intensity. *Comput. Stand. Interfaces* **2025**, *94*, 103990. [CrossRef]
7. Kim, J.; Jin, S.; Park, S.; Park, S.; Han, K. Label-aware Hard Negative Sampling Strategies with Momentum Contrastive Learning for Implicit Hate Speech Detection. In *Proceedings of the Findings of the Association for Computational Linguistics: ACL 2024*; Association for Computational Linguistics: Borough, PA, USA, 2024; pp. 16177–16188.
8. Dehghan, S.; Yanikoğlu, B. Multi-domain Hate Speech Detection Using Dual Contrastive Learning and Paralinguistic Features. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*; ELRA; ICCL: Paris, France, 2024; pp. 11745–11755.
9. Rehman, M.Z.U.; Bhatnagar, A.; Kabde, O.; Bansal, S.; Kumar, D.N. ImphiHateVid: A Benchmark Dataset and Two-stage Contrastive Learning Framework for Implicit Hate Speech Detection in Videos. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 17209–17221.
10. Ghorbanpour, F.; Hangya, V.; Fraser, A. Fine-Grained Transfer Learning for Harmful Content Detection through Label-Specific Soft Prompt Tuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 11047–11061.
11. Nghiem, H.; Daumé, H., III. HateCOT: An Explanation-Enhanced Dataset for Generalizable Offensive Speech Detection via Large Language Models. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024*; Association for Computational Linguistics: Borough, PA, USA, 2024; pp. 5938–5956.
12. Calabrese, A.; Sherborne, T.; Ross, B.; Lapata, M. Compositional Generalisation for Explainable Hate Speech Detection. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 13921–13943.
13. Mothilal, R.K.; Roy, J.; Ahmed, S.I.; Guha, S. Argument-Based Consistency in Toxicity Explanations of LLMs. In *Proceedings of the Findings of the Association for Computational Linguistics: EACL 2026*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 5913–5941.
14. Hu, Y.; Lee, R.K.W. HateXScore: A Metric Suite for Evaluating Reasoning Quality in Hate Speech Explanations. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 4221–4240.
15. Vecchi, L.P.; Britto, A.d.S., Jr.; Paraiso, E.C.; Cruz, R.M. HARM: Learning Hate-Aware Reward Model for Evaluating Natural Language Explanations of Offensive Content. In *Proceedings of the Findings of the Association for Computational Linguistics: EACL 2026*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 4393–4431.
16. Hoang, N.M.; Do, X.L.; Do, D.A.; Vu, D.A.; Anh Tuan, L. ToXCL: A Unified Framework for Toxic Speech Detection and Explanation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2024; pp. 6460–6472.

17. Büyükdemirci, K.; Kucukkaya, I.E.; Ölmez, E.; Toraman, C. JL-Hate: An Annotated Dataset for Joint Learning of Hate Speech and Target Detection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*; ELRA; ICCL: Paris, France, 2024; pp. 9543–9553.
18. Kim, Y.; Park, S.; Han, Y.S. Generalizable Implicit Hate Speech Detection Using Contrastive Learning. In *Proceedings of the 29th International Conference on Computational Linguistics*; ACL Anthology: Gyeongju, Republic of Korea, 2022; pp. 6667–6679.
19. Lee, Y.; Hahn, J.; Ahn, H.; Han, Y.S. AmpleHate: Amplifying the Attention for Versatile Implicit Hate Detection. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 28862–28874.
20. Kim, Y.; Park, S.; Namgoong, Y.; Han, Y.S. ConPrompt: Pre-training a Language Model with Machine-Generated Data for Implicit Hate Speech Detection. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Borough, PA, USA, 2023; pp. 10964–10980.
21. Deng, J.; Zhou, J.; Sun, H.; Zheng, C.; Mi, F.; Meng, H.; Huang, M. COLD: A Benchmark for Chinese Offensive Language Detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2022; pp. 11580–11599.
22. Chen, X.; Wang, S. Pragmatic Inference Chain (PIC) Improving LLMs' Reasoning of Authentic Implicit Toxic Language. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 5826–5841.
23. Yang, J.; Hechtbauer, M.; Khalilov, E.; Brinkmann, E.L.; Schmitt, V.; Feldhus, N. Persona Prompting as a Lens on LLM Social Reasoning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 1152–1170.
24. Piot, P.; Parapar, J. Decoding Hate: Exploring Language Models' Reactions to Hate Speech. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 973–990.
25. Lu, J.; Ma, K.; Wang, K.; Xiao, K.; Lee, R.K.W.; Xu, B.; Yang, L.; Lin, H. Is LLM an Overconfident Judge? Unveiling the Capabilities of LLMs in Detecting Offensive Language with Annotation Disagreement. In *Proceedings of the Findings of the Association for Computational Linguistics: ACL 2025*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 5609–5626.
26. Zhang, T.; Luo, H.; Chuang, Y.S.; Fang, W.; Gaitskell, L.; Hartvigsen, T.; Wu, X.; Fox, D.; Meng, H.; Glass, J. Interpretable Unified Language Checking. *arXiv* **2023**, arXiv:2304.03728.
27. Kang, H.; Qian, T. Implanting LLM's Knowledge via Reading Comprehension Tree for Toxicity Detection. In *Proceedings of the Findings of the Association for Computational Linguistics: ACL 2024*; Association for Computational Linguistics: Borough, PA, USA, 2024; pp. 947–962.
28. He, X.; Zannettou, S.; Shen, Y.; Zhang, Y. You Only Prompt Once: On the Capabilities of Prompt Learning on Large Language Models to Tackle Toxic Content. In *Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP)*; IEEE Computer Society: Washington, DC, USA, 2024; pp. 770–787.
29. Guragain, A.; Poudel, N.; Piryani, R.; Khanal, B. NLPineers@ NLU of Devanagari Script Languages 2025: Hate Speech Detection using Ensembling of BERT-based models. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*; International Committee on Computational Linguistics: Borough, PA, USA, 2025; pp. 320–326.
30. Pérez, J.M.; Miguel, P.; Cotik, V. Exploring Large Language Models for Hate Speech Detection in Rioplatense Spanish. In *Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2025*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 7189–7202.
31. Maab, I.; Haider, U.; Yamagishi, J. Prompt-driven Detection of Offensive Urdu Language using Large Language Models. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 4302–4327.
32. Naseem, U.; Shiwakoti, S.; Shah, S.B.; Thapa, S.; Zhang, Q. GameTox: A Comprehensive Dataset and Analysis for Enhanced Toxicity Detection in Online Gaming Communities. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 440–447.
33. Bokaei, Z.; Magdy, W.; Webber, B. Culture Matters in Toxic Language Detection in Persian. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 9290–9304.
34. Mármol-Romero, A.M.; Sepúlveda-Torres, R.; Saquete, E.; Martín-Valdivia, M.T.; Ureña, L.A. Nuanced Toxicity Detection in Spanish: A New Corpus and Benchmark Study. In *Proceedings of the Findings of the Association for Computational Linguistics: EACL 2026*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 1940–1954.

35. Krasnodębska, A.; Dziewulska, K.; Seweryn, K.; Chrabaszczyk, M.; Kusa, W. Safety of Large Language Models Beyond English: A Systematic Literature Review of Risks, Biases, and Safeguards. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 1003–1034.
36. Naseem, U.; Kashyap, G.S.; Ray, S.K.; Ali, R.; Shabbir, E.; Mohammad, A. Do Large Language Models Reflect Demographic Pluralism in Safety? In *Proceedings of the Findings of the Association for Computational Linguistics: EACL 2026*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 2042–2052.
37. Bell, S.; Meglioli, M.C.; Richards, M.; Sánchez, E.; Ropers, C.; Wang, S.; Williams, A.; Sagun, L.; Costa-jussà, M.R. On the Role of Speech Data in Reducing Toxicity Detection Bias. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 1454–1468.
38. Xiao, Y.; Hu, Y.; Choo, K.T.W.; Lee, R.K.W. ToxiCloakCN: Evaluating Robustness of Offensive Language Detection in Chinese with Cloaking Perturbations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2024; pp. 6012–6025.
39. Wu, J.; Ji, S.; Zhong, K.; Peng, H.; Liu, X.; Wei, W. Enhancing Chinese Offensive Language Detection with Homophonic Perturbation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 22660–22675.
40. Jeon, M.; Jeong, H.; Kim, Y.; Kim, J.; Cho, J.H.; Lee, B.J. K/DA: Automated Data Generation Pipeline for Detoxifying Implicitly Offensive Language in Korean. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2025; pp. 21404–21432.
41. Li, P.; Fillies, J.; Paschke, A. ToxiGAN: Toxic Data Augmentation via LLM-Guided Directional Adversarial Generation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Borough, PA, USA, 2026; pp. 4029–4044.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.