Facilitating Fine-grained Detection of Chinese Toxic Language: Hierarchical Taxonomy, Resources, and Benchmarks

Junyu Lu, Bo Xu, Xiaokun Zhang, Changrong Min, Liang Yang*, Hongfei Lin
School of Computer Science and Technology, Dalian University of Technology, China
(dutljy,kun,11909060)@mail.dlut.edu.cn,(xubo,liang,hflin)@dlut.edu.cn

Abstract

***Disclaimer**: The samples presented by this paper may be considered offensive or vulgar.*

The widespread dissemination of toxic online posts is increasingly damaging to society. However, research on detecting toxic language in Chinese has lagged significantly. Existing datasets lack fine-grained annotation of toxic types and expressions, and ignore the samples with indirect toxicity. In addition, it is crucial to introduce lexical knowledge to detect the toxicity of posts, which has been a challenge for researchers. In this paper, we facilitate the fine-grained detection of Chinese toxic language. First, we build MONITOR TOXIC FRAME, a hierarchical taxonomy to analyze toxic types and expressions. Then, a fine-grained dataset TOXICN is presented, including both direct and indirect toxic samples. We also build an insult lexicon containing implicit profanity and propose Toxic Knowledge Enhancement (TKE) as a benchmark, incorporating the lexical feature to detect toxic language. In the experimental stage, we demonstrate the effectiveness of TKE. After that, a systematic quantitative and qualitative analysis of the findings is given. [1]

1 Introduction

More and more people have acquired information from social media platforms where posts containing toxic language are also rampant. Toxic language is viewed as a rude, disrespectful, or unreasonable comment that is likely to make someone leave a discussion (Dixon et al., 2018). Due to its negative impact on individuals and society, toxic language has been rapidly recognized as an increasing concern (Silva et al., 2016). Recently, researchers have used techniques of natural language processing to detect toxic language, making great progress in many languages (AlKhamissi et al., 2022; Mou et al., 2020; Cao and Lee, 2020; Tekiroglu et al., 2020; Founta et al., 2018; Zhou et al., 2021a; Mathew et al., 2021; Caselli et al., 2021; Hanu and Unitary team, 2020).

In contrast, the relevant research on Chinese toxic language detection has lagged significantly (Jahan and Oussalah, 2023). There are two key issues that have been overlooked. First, existing studies (Deng et al., 2022; Jiang et al., 2022; Zhou et al., 2022) lack a fine-grained annotation of textual toxic types, resulting in hate speech being conflated with general offensive language. Compared to hate speech, general offensive language does not attack groups with special social attributes, and it is just used to emphasize emotions in many cases (Wang et al., 2014). Like Exp. 1 in Table 1, the insult "*fuck*" can be considered as a modal particle to express surprise. Since there is no equivalence between general offensive language and hate speech, it is crucial to determine their boundary conditions (Davidson et al., 2017).

In addition, most studies on toxic Chinese language only concentrate on detecting direct and explicit bias and offense. And they lose sight of indirect expressions including implicit hatred (e.g., stereotype and irony) (ElSherief et al., 2021) and reporting experiences of discrimination (Chiril et al., 2020). Due to the absence of direct swear words, these indirect toxic samples are obviously harder to be filtered (ElSherief et al., 2021). To further illustrate the distinction of several expressions, a few examples are listed in Table 1. Meanwhile, compared to English, Chinese has richer variants of profanity with implicit toxic meaning (Zhang, 2010; Sohn and Lee, 2019), which brings challenges to research on toxic language detection. However, existing insult lexicons fail to cover these terms. An example is "*fairy*" in Exp. 3, which itself is a positive word and is used here to implicitly attack women. Due to the significance of lexical knowledge to detect toxic language (Wiegand et al., 2021;

* Corresponding author
[1] Resources and codes of this paper are available at https://github.com/DUT-lujunyu/ToxiCN.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 16235–16250
July 9-14, 2023

Exp .	Post	Toxic Type	Targeted Group	Expression
1	我靠！我们居然输了。 *What the fuck! I can't believe we lost!*	Offensive	-	Explicitness
2	我一看老黑就想吐。 *I feel like throwing up when I look at n*ggas.*	Hate	Racism	Explicitness
3	小仙女的事你少管。 *Keep your nose out of the fairy's business.*	Hate	Sexism	Implicitness
4	我的朋友说河南人经常偷井盖。 *My friend said Henan people often steal manhole covers.*	Hate	Regional Bias	Reporting

Table 1: Different categories of toxic comment illustration, including *general offensive language* and each hate expression (*explicitness*, *implicitness* and *reporting*).

Hartvigsen et al., 2022), it is important to construct an insult lexicon containing implicit toxic terms.

To fill these gaps, we facilitate fine-grained detection of Chinese toxic language. To distinguish hate speech from general offensive language and analyze the expressions of samples, we first introduce MONITOR TOXIC FRAME, a hierarchical taxonomy. Based on the taxonomy, the posts are progressively divided into diverse granularities as follows: **(I) Whether Toxic, (II) Toxic Type (general offensive language or hate speech), (III) Targeted Group, (IV) Expression Category (explicitness, implicitness, or reporting)**. After taking several measures to alleviate the bias of annotators, we then conduct a fine-grained annotation of posts, including both direct and indirect toxic samples. And TOXICN dataset is presented, which has 12k comments containing *sexism*, *racism*, *regional bias*, and *anti-LGBTQ*.

For the convenient detection of toxic language, we construct an insult lexicon attacking different targeted groups. It contains not only explicit profanities but also implicit words with toxic meanings, such as ironic metaphors (e.g., "*fairy*"). To exploit the lexical feature, we further present a migratable benchmark of Toxic Knowledge Enhancement (TKE), enriching the text representation. In the evaluation phase, several benchmarks with TKE are utilized to detect toxic language, demonstrating its effectiveness. After that, we analyze the experimental results in detail and offer our suggestions for identifying toxic language. The main contributions of this work are summarized as follows:

- We present a hierarchical taxonomy, MONITOR TOXIC FRAME, to progressively explore the toxic types and expressions of samples from diverse granularities.
- Based on the taxonomy, we propose TOXICN, a fine-grained dataset of Chinese toxic language. It divides hate speech from offensive language, including samples with not only direct offense but also indirect expressions.
- We present an insult lexicon, and design a Toxic Knowledge Enhancement benchmark incorporating the lexical feature. We evaluate its performance at different levels and conduct an exhaustive analysis.

2 Related Work

Toxic Language Detection. Toxic language detection is a high-profile task in the field of natural language processing. Recently, most researchers have utilized methods of deep learning based on the pre-trained language model to tackle this problem (Mou et al., 2020; Cao and Lee, 2020; Tekiroglu et al., 2020; Mathew et al., 2021; Zhou et al., 2021a; AlKhamissi et al., 2022). Two re-trained BERT (Devlin et al., 2019), HateBERT (Caselli et al., 2021) and ToxicBERT (Hanu and Unitary team, 2020), have been specifically proposed to detect toxic language. Davidson et al. (2017); Founta et al. (2018); Mathew et al. (2021) attempted to distinguish hate speech from offensive language. ElSherief et al. (2021); Hartvigsen et al. (2022) explored the benchmark of implicit and latent hate speech. Chiril et al. (2020); Pérez-Almendros et al. (2020) considered testing for reporting related to hate speech and behavior. And in the construction of the toxic language dataset, some studies focused on how to improve the reliability of the annotation process to mitigate the subjective bias of annotators (Waseem and Hovy, 2016; Ross et al., 2016; Zeinert et al., 2021; Fortuna et al., 2022).

Linguistic Research of Chinese Toxic Language. Chinese toxic language has been researched extensively in language studies and sociolinguistics.

Work	Source	Scope	Size	Balance	Toxic Type	Expression Category	Implicit Profanity
COLD (Deng et al., 2022)	Zhihu, Weibo	Offensive	37,480	48.1%	✓		
SWSR (Jiang et al., 2022)	Weibo	Hate speech	8,969	34.5%		✓	
CDial-Bias-Utt (Zhou et al., 2022)	Zhihu	Hate speech	13,394	18.9%			
CDial-Bias-Ctx (Zhou et al., 2022)	Zhihu	Hate speech	15,013	25.9%			
TOXICN (ours)	Zhihu, Tieba	Offensive and hate speech	12,011	53.8%	✓	✓	✓

Table 2: Summary of Simplified Chinese toxic language datasets in terms of *Source*, *Scope*, *Size*, toxic class ratio (*Balance*), and the inclusion of *Toxic Type*, *Expression category*, and the construction of the lexicon containing *Implicit Profanity*.

Zhang (1994) analyzed Chinese vernacular novels and summarized the common rhetorical methods of offensive language. According to Wang and Liu (2009); Li et al. (2020), insults are more easily expressed through variants. Due to the lack of morphological markers, authors often make sentences based on language sense and consensus (Zhang, 2004), expressing hatred more concisely compared to Indo-European (Zhang, 2005). Zhang (2010); Sohn and Lee (2019) compared insults in English and Chinese and discovered that Chinese has a richer variety of profanity due to its unique culture and linguistics. These linguistic features bring challenges to Chinese toxic detection. Recently, some Chinese toxic language datasets have been constructed (Deng et al., 2022; Jiang et al., 2022; Zhou et al., 2022). However, they fails to separate hate speech from general offensive language, and overlooks toxic samples containing indirect expressions. Besides that, it lacks the construction of the lexicon containing implicit profanities. In this work, we fill these gaps to facilitate fine-grained detection of Chinese toxic language. Here we list Table 2 to compare these studies with our TOXICN.

3 Dataset Construction

3.1 Overview

In this section, we describe the construction of TOXICN dataset. We first introduce the process of data collection and filtering. Then, MONITOR TOXIC FRAME is presented as the guideline for labeling. After adopting several measures to mitigate biases in the process of labeling, we implement a hierarchical fine-grained annotation based on the frame. The Inter-Annotator Agreement (IAA) of each individual granularity is explored. Finally, related statistics of TOXICN are shown.

3.2 Data Collection and Filtering

To avoid a high degree of homogenization of data, we crawl the published posts from two public online media platforms, *Zhihu* and *Tieba*. Both platforms are representative of local users in China and have active communities about specific topics. Due to the filtering mechanism of the websites, the proportion of posts containing toxic language is relatively sparse (Deng et al., 2022). Thus, we first limit the scope of crawled data under several sensitive topics, including *gender*, *race*, *region*, and *LGBTQ*, which are easily debated on the Internet. And then, we list some keywords for each topic and utilize them to extract a total of 15,442 comments without replies. We remove the samples where the text is too brief to have actual semantics, such as phrases consisting of only inflections and auxiliaries. And dirty data is also deleted, including duplicated samples and irrelevant advertisements. In the end, 12,011 comments are retained[2].

In the stage of data cleaning, we focus on normalizing the unique form of expression in web text adopted from Ahn et al. (2020). The extra newline characters and spaces in the original text are also deleted. To prevent privacy leakage, we desensitize the data, filtering out *@USERs*, attached links, and pictures. Due to the possibility of carrying important emotional information (Mathew et al., 2021), we reserve emojis for toxic detection.

[2]We also attempted to crawl posts from *Weibo* referenced (Deng et al., 2022), however, due to the filtering mechanism, there is no guarantee that sufficient samples will be obtained under each topic, such as "*race*" and "*LGBTQ*", resulting in a relatively homogeneous crawl. Therefore, we finally choose *Zhihu* and *Tieba* as our data sources.

3.3 Data Annotation

3.3.1 Monitor Toxic Frame

To determine the downstream subtasks and establish the finalized guidelines for labeling, a standardized taxonomy is needed. Here we build a hierarchical taxonomy called MONITOR TOXIC FRAME (in Figure 1). It has three levels including four diverse aspects to identify toxic content and further in-depth analysis. The specific implementations are as follows:

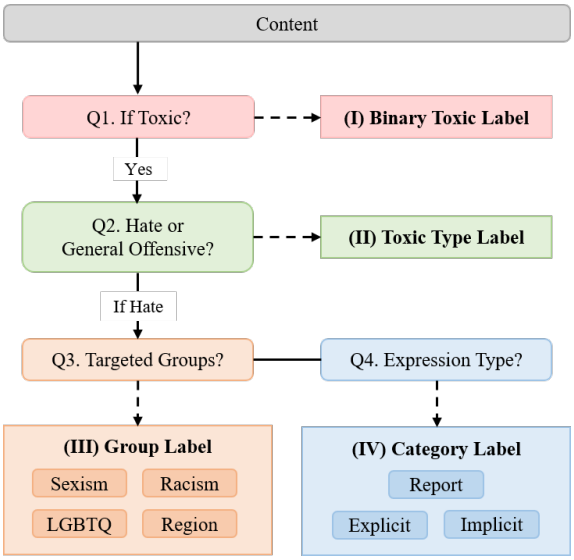

Figure 1: MONITOR TOXIC FRAME illustration. The framework introduces four questions to determine whether a comment is general offensive language or hate speech, and further analyzes the attacked group and expression type.

Toxic Identification. The first level of our framework is a binary annotation to determine whether the comment contains toxic language, which is the foundation of subsequent labeling. In this work, general offensive language and hate speech are highlighted.

Toxic Type Discrimination. The second level is to distinguish general offensive language and hate speech. Based on Waseem and Hovy (2016) and Fortuna and Nunes (2018), we list several criteria for identification of hate speech: 1) attacking specific groups, or 2) inciting others to hate minorities, or 3) creating prejudice, rejection, or disgust for minorities based on stereotypes and distorted facts, or 4) using sarcasm or humor to ridicule groups, despite the publisher may not be malicious. In contrast, general offensive language is not insulting to targets with specific social attributes compared to hate speech (Davidson et al., 2017).

Targeted Group and Expression Type Detection. In the third level, we further explore the targeted group and expression type of toxic language. If the content contains hate speech, its characteristics of the target group are specified, including *sexism*, *racism*, *regional bias*, and *anti-LGBTQ*. Since multi-class groups may be attacked in a text, this task is categorized as multi-label classification. Meanwhile, we determine the categories of toxic expressions, containing *explicitness*, *implicitness*, and *reporting*. In these expressions, 1) explicitness is obviously offensive, inciting, or prejudiced against minority groups, and 2) implicitness refers to the subtle or even humor expression without strong exclusion, such as microaggressions and stereotyping (Hartvigsen et al., 2022), and 3) reporting only documents or condemns experience of attack and discrimination against minorities, and the publisher does not express hatred and prejudice (Chiril et al., 2020). As the expression category of general offensive language is necessarily explicit, we focused on the expressions in hate speech. This granularity is set as multi-classification.

3.3.2 Mitigating Bias

The subjective biases of the annotators negatively impact the quality of the dataset (Waseem and Hovy, 2016). Therefore, it is significant to mitigate these biases during the design and construction of annotations. For this purpose, we adopt the following measures: We first guarantee the diversity of annotators in terms of background information, including gender, age, race, region, and study. All participants major in linguistics and have been systematically trained. The demographics of annotators are shown in Table 3. Then, we make a progressive analysis of the toxic content contained in the crawled posts, and initially determine the labeling rules for various granularities. After a couple of iterations of small-scale annotation tests and discussions of edge cases, the final criteria are established.

Characteristic	Demographics
Gender	5 male, 4 female
Age	5 age $<$ 25, 4 age $\geq$ 25
Race	6 Asian, 3 others
Region	From 5 different provinces
Education	2 BD, 4 MD, 3 Ph.D.

Table 3: Annotators demographics.

Topic	N-Tox.	Tox.	Toxic Category					Total	Avg. *L*
			Off.	Hate	H-exp.	H-imp.	H-rep.		
Gender	1,805	2,153	316	1,837	1,055	693	89	3,958	35.26
Race	1,602	2,084	229	1,855	1,041	711	103	3,686	36.93
Region	1,222	1,148	82	1,066	172	292	602	2,370	40.26
LGBTQ	921	1,076	189	887	469	299	119	1,997	44.96
Total	5,550	6,461	816	5,645	2,737	1,995	913	12,011	38.37

Table 4: Basic statistics of TOXICN, listing the number of non-toxic (*N-Tox.*) and toxic (*Tox.*) comments, containing general offensive language (*Off.*) and each hate expression categories (including explicitness (*H-exp.*), implicitness (*H-imp.*) and reporting (*H-rep.*)). And *Avg. L* is the average length of samples.

3.3.3 Annotation Procedure

The annotation procedure consists of two steps: pseudo labeling and main manual annotation. Meanwhile, the initial construction of the insult lexicon is implemented.

Pseudo Labelling. To reduce the burden of manual annotation, we retrieve the samples containing insults, most of which are obviously toxic. Specifically, we first build an original lexicon of explicit profanity words, integrating two existing profanity resources, including HateBase[3], the world's largest collaborative and regionalized repository of multilingual hate speech, and SexHateLex[4] (Jiang et al., 2022), a large Chinese sexism lexicon. Then, an iterative approach is employed to match out profanity-laced comments using regular expressions. The swearwords contained in these samples that are not in the lexicon are further collected. We assign a toxic pseudo-label for each sentence containing insults. After several iterations, the remaining samples are directly pseudo-labeled as non-toxic. The statistics illustrate that this method is simple and effective, correctly separating about 50% of toxic samples from ToxiCN. See Table D3 from Appendix D for a more detailed report.

Main Manual Annotation. Based on MONITOR TOXIC FRAME, we implement the main annotation of TOXICN. Most samples pseudo-labeled as toxic are directly categorized as general offensive language or explicit toxic language. Afterwards, due to the low frequency variants of insults and implicit toxicity expressions, the remaining pseudo-labeled as non-toxic samples have to be re-annotated in a hierarchical manner. Meanwhile, the implicit insults contained in these samples are added to the previous profanity list. We utilize the open source text annotation tool Doccano[5] to facilitate the labeling process. Each comment is labeled by at least three annotators and a majority vote is then used to determine the final label. After annotation, we explore the Inter-Annotator Agreement (IAA) of TOXICN, and Fleiss' Kappa of each hierarchy is shown as Table 5.

If Toxic	Toxic Type	Targeted	Expression
0.62	0.75	0.65	0.68

Table 5: Fleiss' Kappa for different granularities.

3.4 Data Description

In the data analysis phase, we first describe the dataset from the topic of comments. The basic statistics of TOXICN are shown in Table 4. We note that there is a sample imbalance between different categories of toxic samples. Specifically, the *Off.* class represents only 6.8% of the overall dataset. Because the data distribution reflects the true situation of the platforms (Mathew et al., 2021), we do not apply additional treatment to the imbalance. In addition, since a single case from a topic may attack multi-class groups, we further record the sample size of hate speech against various target categories. From Table 6, we can see that the distribution of expressions is different for each group. For example, most samples containing regional bias are reporting, which is uncommon in other categories. More statistical details are listed in Appendix D.

Group	H-exp.	H-imp.	H-rep.	Total
Sexism	1,259	887	156	2,302
Racism	1,149	660	65	1,874
RGN. B.	295	384	610	1,289
Anti-L+	671	383	121	1,075

Table 6: Sample size of various hate expressions for each attacked group label. *RGN. B.* refers to regional bias and *Anti-L+* is short for anti-LGBTQ.

[3]https://hatebase.org/

[4]https://zenodo.org/record/4773875

[5]https://github.com/doccano/doccano

4 Insult Lexicon

We divide the insult lexicon built in the process of annotation into five categories according to the attacking object. The lexicon includes sexism, racism, regional bias, anti-LBGTQ, and general swearwords, referring to the swear words that can be used to offend any group. The final dictionary contains 1,032 terms, including not only explicit vulgar words but also implicit insults. In addition, as the Internet generates a wealth of new insults every year, it is vital to explore their origins. Therefore, for the convenience of the follow-up research, we further analyze the rules for the derivation of Internet swear words from the proposed insult lexicon. Specifically, we briefly summarize them in terms of both surface features and their actual meaning. More related terminology notes and examples of profanity are illustrated in Appendix C.

Surface Features. To circumvent the filtering mechanism, netizens change the original insults and create new variants with similarities in glyphs and pronunciation (Chen, 2012; Zhang, 2010), which are called *deformation* and *homophonic*, respectively. In addition, Chinese characters are at times replaced with other language codes in some profanities (Li et al., 2020), creating *code-mixing words* or *abbreviations*.

Actual Meaning. Internet users often introduce implicit terms to attack or defame the target groups, including usage of *metaphor* and *irony* (Chen, 2012). Besides that, some *borrowed words* also contain specific prejudices, which are used in implicit toxic comments (Shi, 2010). Compared to variants based on surface features, these terms with deep semantics have to be detected with background knowledge.

5 Methodology

In view of the significance of lexical knowledge to detect toxic language, we propose a migratable benchmark of Toxic Knowledge Enhancement (**TKE**), incorporating the lexical feature to enrich the original sentence representation. The illustration of TKE is shown in Figure 2.

For a given sentence $S = \{x_1, x_2, ..., x_n\}$, each token x_i is embedded as $w_i \in \mathbb{R}^d$, which is a vector representation in d-dimensional space. Inspired by Zhou et al. (2021a), we design a toxic embedding to introduce lexical knowledge. Specifically, we first employ the n-gram to determine whether x_i is a subword of an insult, and if so, its attacked group is further indicated. Then, we randomly initialize the group category representation as $C = (c_0, c_1, ..., c_m)$, where $c_i \in \mathbb{R}^d$, c_0 refers to non-toxic term and m is the number of categories of the insult lexicon. In this work, $m = 5$. Based on the c_i, we further propose the definition of toxic embedding t_i of x_i:

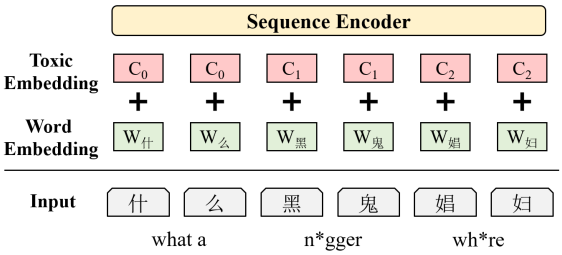

Figure 2: Toxic Knowledge Enhancement (TKE) illustration. Here we set the category representations of non-toxic terms, racist terms, and sexist terms as C_0, C_1, and C_2, respectively.

$$t_i = \begin{cases} c_o, & \text{if } x_i \text{ is non-toxic.} \\ c_j, & \text{if } x_i \text{ is from the } j^{th} \text{ category.} \end{cases} \quad (1)$$

Since the element-wise addition of multiple linear vector representations is an efficient way to fully incorporate various information (Mikolov et al., 2013), we utilize this method to integrate toxic and word embedding. The enhanced representation of x_i is $w'_i = w_i + \lambda t_i$, where $\lambda \in [0, 1]$ is a weighting coefficient to control the ingestion of toxic knowledge. The ultimate sentence embedding of S is $\{w'_1, w'_2, ..., w'_n\}$, which is the input of the connected sequence encoder. Due to its convenience, TKE can be migrated to any pre-trained language model (PLM).

6 Experiments

6.1 Baselines

Here we introduce the baselines of experiments. Several PLMs are utilized as encoders as follows. And we use a fully-connected layer as the classifier for several subtasks.

BiLSTM. This method employs the word vector of Tencent AI Lab Embedding[6], a static word vector with 200-dimensional features, and integrates contextual information using BiLSTM. We concatenate the last hidden states from both forward

[6]https://ai.tencent.com/ailab/nlp/zh/embedding.html

	Toxic Identification			Toxic Type			Targeted Group			Expression Category		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
BTC	64.2	53.0	45.9	-	-	-	-	-	-	-	-	-
BiLSTM	$73.7_{0.6}$	$72.7_{0.4}$	$72.9_{0.4}$	$77.7_{2.2}$	$70.4_{1.2}$	$73.7_{0.5}$	$61.1_{1.2}$	$64.4_{0.7}$	$62.2_{0.5}$	$49.7_{1.6}$	$48.6_{1.1}$	$48.0_{1.0}$
BiLSTM*	$75.4_{0.5}$	$74.8_{0.5}$	$74.9_{0.4}$	$79.2_{2.0}$	$69.6_{1.4}$	$73.6_{0.4}$	$58.8_{0.9}$	$68.6_{1.3}$	$62.8_{0.7}$	$51.2_{1.9}$	$56.8_{1.7}$	$53.5_{1.3}$
BERT	$80.0_{0.2}$	$79.7_{0.2}$	$79.7_{0.2}$	$\mathbf{82.8}_{0.9}$	$73.1_{1.0}$	$77.3_{0.4}$	$71.1_{0.9}$	$71.9_{1.0}$	$72.2_{0.4}$	$\mathbf{55.9}_{1.1}$	$56.4_{1.3}$	$55.3_{1.1}$
BERT*	$80.4_{0.3}$	$80.2_{0.3}$	$80.0_{0.3}$	$80.2_{1.3}$	$75.2_{0.9}$	$77.3_{0.2}$	$\mathbf{73.3}_{1.2}$	$72.6_{0.9}$	$72.6_{0.4}$	$53.5_{2.0}$	$60.9_{0.5}$	$55.9_{0.9}$
RoBERTa	$80.8_{0.2}$	$80.2_{0.3}$	$80.3_{0.3}$	$80.5_{0.9}$	$74.6_{0.5}$	$77.3_{0.3}$	$71.8_{1.4}$	$73.9_{1.1}$	$72.6_{0.5}$	$54.2_{1.2}$	$58.7_{1.1}$	$55.8_{0.6}$
RoBERTa*	$\mathbf{80.9}_{0.3}$	$\mathbf{80.5}_{0.3}$	$\mathbf{80.6}_{0.3}$	$79.8_{1.8}$	$\mathbf{76.1}_{1.2}$	$\mathbf{77.7}_{0.4}$	$72.5_{0.9}$	$\mathbf{74.0}_{1.0}$	$\mathbf{73.0}_{0.5}$	$54.4_{1.6}$	$\mathbf{61.3}_{1.2}$	$\mathbf{56.8}_{0.4}$

Table 7: Evaluation of each subtask. Results show the mean and s.d. (subscript) of P, R, and F_1, where BTC denotes Baidu Text Censor, * refers to the introduction of TKE to the baseline, and the **bold** score represents the best obtained values. Because BTC is an online API with no training required, we use it to perform the toxic identification of all the samples in TOXICN.

and backward directions to obtain the final sentence embedding.

BERT (Devlin et al., 2019) and **RoBERTa** (Liu et al., 2019). The two most commonly used Chinese transformer-based PLMs, `bert-based-chinese`[7] and `roberta-base-chinese`[8], are used as benchmarks. In the experiment, the pooled output of the encoder is utilized as the input of the connected classifier.

Besides the above deep learning based methods, we also evaluate the performance of **Baidu Text Censor**[9], an online API to identify toxic content. Due to the function limit, we only utilize it in the first subtask of binary toxic identification.

6.2 Implementation

We employ the widely used metrics of weighted precision (P), recall (R), and F_1-score (F_1) to evaluate the performance of models. Weighted cross entropy is utilized to address the problem of category imbalances, and the optimizer is AdamW. An early stopping mechanism is applied in the training phase. All the samples in TOXICN are split into a training set and a test set with a ratio of 8:2. We fine-tune the baselines and reserve the best performing models and hyperparameters on the test set, and the same experiments are repeated 5 times by changing the random seeds for error reduction. All experiments are conducted using a GeForce RTX 3090 GPU. More details are shown in Appendix E.

6.3 Results and Discussions

In this section, we present our experimental results and progressively analyze the following three questions, respectively:

RQ1: Performance of Different Subtasks. We evaluate the performance of each baseline and the contribution of TKE at different granularities of toxic language detection. The experimental results are shown in Table 7. From the results, we can observe that:

(1) Compared with Baidu Text Censor, deep learning based methods achieved better performance. A plausible explanation is the filtering mechanism of the online API mainly depends on the keyword dictionary. Therefore, it cannot effectively detect the toxicity of sentences containing indirect expressions of hate speech. In addition, the performance of the pre-trained language model based on dynamic word representation (e.g., BERT, RoBERTa) is much better than on static embedding. And in these baselines, RoBERTa is the most effective on several subtasks.

(2) Overall, the models introducing TKE have improved the performance of several subtasks, illustrating the effectiveness of representation incorporating toxic lexical knowledge. Among them, TKE leads to the greatest enhancement in the detection of expression category, with an average improvement of 2.7% of each baseline. This result shows that lexical information can improve the ability of the model to detect toxic language in different expressions. Meanwhile, we also find TKE does not bring significant improvement in toxic type discrimination. This is because many insults are widely contained in both general offensive language and hate speech. Therefore, the introduction of toxic embedding does not distinguish well between these two kinds of speech.

RQ2: Detection of Each Toxic Subtype. The complementary experiment is conducted to further

[7]https://huggingface.co/bert-base-chinese
[8]https://huggingface.co/hfl/chinese-roberta-wwm-ext
[9]https://ai.baidu.com/tech/textcensoring

evaluate the performance of models to identify the toxic content with different expressions. Specifically, we utilize the optimal trained models for the subtask of toxic identification to detect the samples in the test set. After that, we separately analyze the accuracy of sentences labeled as non-toxic and each expression category. The results are shown in Figure 3.

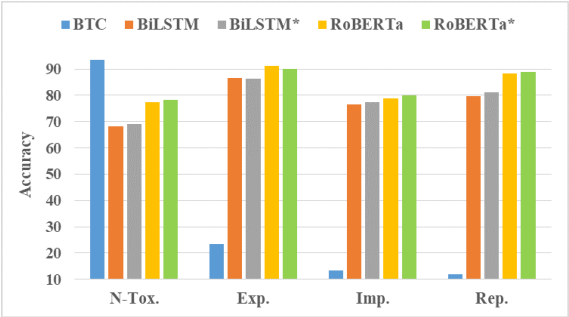

Figure 3: Accuracy towards samples with different expressions, containing non-toxicity (*N-Tox.*), explicitness (*Exp.*), implicitness (*Imp.*), and reporting (*Rep.*).

Based on the result, it is noteworthy that:

(1) Compared to explicit toxic language, the accuracy of implicit expression is significantly lower, with a difference of around 10%. The reason is that many samples containing implicit toxicity are mistakenly classified as non-toxic due to the absence of explicit insults.

(2) In spite of more training data, the performance of models to detect implicit toxicity is worse than reporting about 5%. This is because the reporting contains references to actors such as "*he/she said...*" which support decisions of models.

(3) The introduction of TKE increases the accuracy of the model for implicit toxicity and reporting samples. It illustrates that the implicit lexcial knowledge enhances the ability of models to detect toxic samples with indirect expressions.

RQ3: Error Analysis. For more insight into the performance of TKE, we perform a manual inspection of the set of samples misclassified by all the models. Two main types of errors are summarized. Here we list the following two samples in the set for illustrative purposes:

Exp. 1 北京高考400分上清华 — *Toxic*
(It is sufficient to get into THU with a NEMT result of 400 points in Beijing.)

Exp. 2 他以前发帖说过自己是 **txl**。 — *Non-Toxic*
*(He has posted before that he is **gay**.)*

Type I error refers to sentences annotated as *toxic*, but classified as *non-toxic* by the models. This kind of error usually occurs in the detection of samples containing implicit bias, caused by a lack of background information on the semantic level. Like Exp. 1, supported by external knowledge, including 400 is a relatively low score on the NEMT with a full score of 750, and THU is one of the best universities in China, it can be known that the publisher uses a fake message to express implicit regional bias against Beijing.

Type II error denotes to instances labeled as *non-toxic*, while detected as *toxic*. The samples with this error usually contain toxic token-level markers, such as swear words and pronouns of minority groups. The training data with these markers is often labeled as toxic language, leading to spurious associations in the models (Zhou et al., 2021b; Ramponi and Tonelli, 2022). Like Exp. 2, where "txl" (meaning "*gay*") causes models to make decisions based only on statistical experience rather than incorporating context information.

From the error analysis, we note that it remains a challenge to integrate richer external knowledge without reducing spurious biases of models. In future work, we will further explore methods of introducing knowledge enhancement for toxic language detection.

7 Conclusion and Future Work

Due to the rampant dissemination of toxic online language, an effective detection mechanism is essential. In this work, we focus on Chinese toxic language detection at various granularities. We first introduce a hierarchical taxonomy MONITOR TOXIC FRAME to analyze the toxic types and expressions. Based on the taxonomy, we then propose a fine-grained annotated dataset TOXICN, including both direct and indirect toxic samples. Due to the significance of lexical knowledge for the detection of toxic language, we build an insult lexicon and present a benchmark of Toxic Knowledge Enhanced (TKE), enriching the representation of content. The experimental results show the effectiveness of TKE on toxic language detection from different granularities. After an error analysis, we suggest that both knowledge introduction and bias mitigation are essential. We expect our hierarchical taxonomy, resources, benchmarks, and insights to help relevant practitioners in the identification of toxic language.

Limitations

Despite the fact that some measures have been implemented to minimize bias in labeling, we are still explicitly aware that our dataset may contain mislabeled data due to differences in the subjective understanding of toxic language by the annotators. In addition, due to the limitation of data coverage, the samples in our dataset are predominantly in Simplified Chinese, with very few samples in Traditional Chinese, as discussed in Section D.5. Meanwhile, as shown in the error analysis in Section 6.3, our benchmark of Toxic Knowledge Enhanced is not practical for all types of toxic comments, lacks sufficient background knowledge, and can easily lead to spurious associations.

For reasons of intellectual property, we only capture the comments rather than the full text, which affects the actual semantics of the sentence to some extent. Besides that, non-textual features are not taken into account in this work, such as images and meta information about publishers. In future work, we will further research span-level and multi-modal toxic language detection.

Ethics Statement

We strictly follow the data use agreements of each public online social platform and double-check to ensure that there is no data relating to user privacy. The opinions and findings contained in the samples of our presented dataset should not be interpreted as representing the views expressed or implied by the authors. We hope that the benefits of our proposed resources outweigh their risks. All resources are for scientific research only.

Acknowledgment

This research is supported by the Natural Science Foundation of China (No. 62076046, 62006034). We would like to thank all reviewers for their constructive comments.

References

Hwijeen Ahn, Jimin Sun, Chan Young Park, and Jungyun Seo. 2020. Nlpdove at semeval-2020 task 12: Improving offensive language detection with cross-lingual transfer. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 1576–1586. International Committee for Computational Linguistics.

Badr AlKhamissi, Faisal Ladhak, Srinivasan Iyer, Veselin Stoyanov, Zornitsa Kozareva, Xian Li, Pascale Fung, Lambert Mathias, Asli Celikyilmaz, and Mona Diab. 2022. ToKen: Task decomposition and knowledge infusion for few-shot hate speech detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2109–2120, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Rui Cao and Roy Ka-Wei Lee. 2020. Hategan: Adversarial generative-based data augmentation for hate speech detection. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6327–6338. International Committee on Computational Linguistics.

Tommaso Caselli, Valerio Basile, Mitrovic Jelena, Granitzer Michael, et al. 2021. Hatebert: Retraining bert for abusive language detection in english. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25. Association for Computational Linguistics.

Wangdao Chen. 2012. *Rhetoric introduction*. Fudan University Press.

Patricia Chiril, Véronique Moriceau, Farah Benamara, Alda Mari, Gloria Origgi, and Marlène Coulomb-Gully. 2020. An annotated corpus for sexism detection in french tweets. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 1397–1403. European Language Resources Association.

Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017, Montréal, Québec, Canada, May 15-18, 2017*, pages 512–515. AAAI Press.

Jiawen Deng, Jingyan Zhou, Hao Sun, Chujie Zheng, Fei Mi, Helen Meng, and Minlie Huang. 2022. COLD: A benchmark for Chinese offensive language detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11580–11599, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Emily Dinan, Samuel Humeau, Bharath Chintagunta, and Jason Weston. 2019. Build it break it fix it for

dialogue safety: Robustness from adversarial human attack. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4536–4545. Association for Computational Linguistics.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018*, pages 67–73. ACM.

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. 2021. Latent hatred: A benchmark for understanding implicit hate speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 345–363. Association for Computational Linguistics.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Paula Fortuna, Mónica Domínguez, Leo Wanner, and Zeerak Talat. 2022. Directions for NLP practices applied to online hate speech detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11794–11805. Association for Computational Linguistics.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Comput. Surv.*, 51(4):85:1–85:30.

Antigoni-Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018, Stanford, California, USA, June 25-28, 2018*, pages 491–500. AAAI Press.

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

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 3309–3326. Association for Computational Linguistics.

Md Saroar Jahan and Mourad Oussalah. 2023. A systematic review of hate speech automatic detection using natural language processing. *Neurocomputing*, pages 126–232.

Aiqi Jiang, Xiaohan Yang, Yang Liu, and Arkaitz Zubiaga. 2022. SWSR: A chinese dataset and lexicon for online sexism detection. *Online Soc. Networks Media*, 27:100–182.

Bin Li, Yan Dou, Yingting Cui, and Yuqi Sheng. 2020. Swearwords reinterpreted: New variants and uses by young chinese netizens on social media platforms. *Pragmatics*, 30(3):381–404.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

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 *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 14867–14875. AAAI Press.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Guanyi Mou, Pengyi Ye, and Kyumin Lee. 2020. SWE2: subword enriched and significant word emphasized framework for hate speech detection. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 1145–1154. ACM.

Carla Pérez-Almendros, Luis Espinosa Anke, and Steven Schockaert. 2020. Don't patronize me! an annotated dataset with patronizing and condescending language towards vulnerable communities. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 5891–5902. International Committee on Computational Linguistics.

Alan Ramponi and Sara Tonelli. 2022. Features or spurious artifacts? data-centric baselines for fair and robust hate speech detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle,*

WA, United States, July 10-15, 2022, pages 3027–3040. Association for Computational Linguistics.

Bjorn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky, and Michael Wojatzki. 2016. Measuring the reliability of hate speech annotations: The case of the european refugee crisis. In *3rd Workshop on Natural Language Processing for Computer-Mediated Communication/Social Media*, pages 6–9. Ruhr-Universitat Bochum.

Chunhong Shi. 2010. Web language as a language variety and a linguistic issue. *Applied Linguistics*, (3):70–80.

Leandro Araújo Silva, Mainack Mondal, Denzil Correa, Fabrício Benevenuto, and Ingmar Weber. 2016. Analyzing the targets of hate in online social media. In *Proceedings of the Tenth International Conference on Web and Social Media, Cologne, Germany, May 17-20, 2016*, pages 687–690. AAAI Press.

Hajung Sohn and Hyunju Lee. 2019. MC-BERT4HATE: hate speech detection using multi-channel BERT for different languages and translations. In *2019 International Conference on Data Mining Workshops, ICDM Workshops 2019, Beijing, China, November 8-11, 2019*, pages 551–559. IEEE.

Serra Sinem Tekiroglu, Yi-Ling Chung, and Marco Guerini. 2020. Generating counter narratives against online hate speech: Data and strategies. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1177–1190. Association for Computational Linguistics.

Junjie Wang and Haiyan Liu. 2009. A brief discussion on internet criticism. *Journal of Language and Literature Studies*, (10):152–153.

Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, and Amit P. Sheth. 2014. Cursing in english on twitter. In *Computer Supported Cooperative Work, CSCW '14, Baltimore, MD, USA, February 15-19, 2014*, pages 415–425. ACM.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the Student Research Workshop, SRW@HLT-NAACL 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 88–93. The Association for Computational Linguistics.

Michael Wiegand, Josef Ruppenhofer, and Elisabeth Eder. 2021. Implicitly abusive language - what does it actually look like and why are we not getting there? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 576–587. Association for Computational Linguistics.

Philine Zeinert, Nanna Inie, and Leon Derczynski. 2021. Annotating online misogyny. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3181–3197. Association for Computational Linguistics.

Cuanbiao Zhang. 2004. Different courses of development between chinese and european poetries: A perspective from the nature of chinese characters. *Journal of Pla University of Foreign Languages*, 27(6):77–82.

Cuanbiao Zhang. 2005. A probe into the differences between chinese and western cultures from the perspective of their swearwords. *Training and Research-Journal of Hubei College of Education*, 22(4):46–49.

Tingxing Zhang. 1994. A first look at folk placement honorifics. *Folklore Studies*, (03):30–35.

Yimin Zhang. 2010. On the phonetic features of cursing words in chinese and english. *Contemporary Rhetoric*, 26(4):1–13.

Jingyan Zhou, Jiawen Deng, Fei Mi, Yitong Li, Yasheng Wang, Minlie Huang, Xin Jiang, Qun Liu, and Helen Meng. 2022. Towards identifying social bias in dialog systems: Framework, dataset, and benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3576–3591, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xianbing Zhou, Yang Yong, Xiaochao Fan, Ge Ren, Yunfeng Song, Yufeng Diao, Liang Yang, and Hongfei Lin. 2021a. Hate speech detection based on sentiment knowledge sharing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7158–7166.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah A. Smith. 2021b. Challenges in automated debiasing for toxic language detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3143–3155. Association for Computational Linguistics.

A Sample

We adopt JSON file to store TOXICN dataset, which is a mainstream coding specification to facilitate machine-readable. The construction of data is *Sample = (ID, Platform, Topic, Text, Toxic, Hate, [Group], [Expression])*, where *Toxic* and *Hate* denote whether the sentence contains toxic language or hate speech, respectively. And if it is not biased, *Group* and *Expression* are set to empty. Here we provide two samples in Figure A1.


```
[
    {
        "ID": 1,
        "Platform": "Zhihu",
        "Topic": "Race",
        "Text": "中国男人不会接盘跟老黑交往过的女人。",
        "Toxic": "True",
        "Hate": "True",
        "Group": ["Sexism", "Racism"],
        "Expression": "Implicitness"
    },
    {
        "ID": 2,
        "Platform": "Baidu",
        "Topic": "Gender",
        "Text": "惊了，当初看她是因为手和脚特别好看",
        "Toxic": "False",
        "Hate": "False",
        "Group": [],
        "Expression": ""
    }
]
```


Figure A1: Two samples of TOXICN.

B Details of Annotation

We introduce Fleiss' Kappa (Fleiss, 1971) as the measure of agreement, which works for any number of raters. For the i-th sample, the calculation of Kappa P_i is as follows:

$$P_i = \frac{1}{n(n-1)} \left[\left(\sum_{j=1}^{k} n_{i,j}^2 \right) - n \right], \quad (2)$$

where n and k denote the number of raters and categories, respectively, and $n_{i,j}$ represents the number of raters who assigned the i-th sample to the j-th category.

In the process of annotation, if the root category is wrong, labels of its subcategories, which are annotated by minorities, will be discarded. Then, new raters will be introduced to label this sample. And the Kappa of this sample is recalculated. Based on the IAA shown in Table 5, the most disagreement stems from the phase of "Toxic Identification" and the Kappa is 0.62, caused by implicit hate speech (*H-imp*) containing language techniques like humor. Although we regard these samples as toxic in the rules, some annotators believe that, due to subjective reasons, their toxicity intensity is insufficient to classify them as toxic. And in the "Targeted Group" with a Kappa of 0.65, "*sexism*" and "*anti-LGBTQ*" can also easily be confused.

C Derivative Rules of insults

In this section, we further explain the term in Section 4 and list several insults shown in Table C1, presenting their morphologies to analyze the literal and flexible derived meanings.

Deformation. Since Chinese characters are pictographs, they will be given meanings containing specific emotions by separating and combining with individual characters (Chen, 2012). An example is "默" (meaning "*silence*"), whose glyph consists of "黑" (meaning "*black*") and "犬" (meaning "*dog*"), implicitly expressing the distaste for the black community.

Homophonic. Like English, a new word with a similar pronunciation can be substituted for the original word, resulting in the creation of a different semantics (Zhang, 2010). For instance, netizens always substitute "满" (meaning "*Manchu*") for "蛮" (meaning "*barbarians*"), both of which are pronounced similarly to "*man*".

Irony. Positive words is sometimes ironically used to achieve the effect of insults, which is often reflected in old words with new meanings (Fortuna and Nunes, 2018). Like "仙女" (meaning "*fairy*"), what was originally a gentle and kind image is implied to be a rude and impolite "*shrew*".

Abbreviation. Shortening and contracting sensitive words will make expressions more concise and clear (Chen, 2012). An example is "*txl*", where each letter is the pronounced initials of "同", "性", and "恋", respectively, meaning "*gay*".

Metaphor. Internet users often degrade their attacking targets into something sarcasm, such as animals, in order to insult them (Zhang, 2010). In the term "蠢驴" (meaning "*silly donkey*"), the publisher compares men to donkeys, transmitting aggression against others.

Code Mixing. To emphasize the tone, non-Chinese language codes are widely mixed in the

Term	Literal Meaning	Composition	Actual Meaning	Category
默(mò)	silence	黑(hēi) 犬(quǎn) → black dog	n*gger	racial
南(nán) 满(mǎn)	South Manchu	南满 → 南蛮(mán)	southern barbarians	regional
蠢驴	silly donkey	-	foolish people	general
txl	txl	txl → 同(tóng) 性(xìng) 恋(liàn)	gay	anti-L+
ni哥(gē)	ni brother	ni+ger → n*gger	n*gger	racial
小(xiǎo) 仙(xiān) 女(nǚ)	fairy	-	shrew	sexual
凯(kǎi) 勒(lè) 奇(qí)	Kalergi	-	Kalergi Plan	racial

Table C1: Example illustration of Chinese insults. Among them, *Composition* means the structure and formation of these words, while it is the lexical foundation to derive the *Actual Meaning*.

Topic	Keywords
Gender	性别歧视, 性别偏见, 男权主义, 女权主义, 性别对立, 父权, 家庭主妇 *Sexism, Gender Bias, Masculinity, Feminism, Gender Dichotomy, Patriarchy, Housewife*
Race	种族歧视, 人种, 黑种人, 白种人, 混血儿, 少数民族, 血统, 肤色, 亚裔 *Racism, Ethnic, Black Race, White Race, Mixed-Blood, Ethnic Minority, Bloodline, Skin Color, Asian*
Region	地域歧视, 非洲, 东南亚, 上海, 北京, 广州, 南方人, 北方人 *Regional Discrimination, Africa, Southeast Asia, Shanghai, Beijing, Guangzhou, Northerners, Southerners*
LGBTQ	反同性恋, 异性恋, 同性恋, 女同性恋, 男同性恋, 双性向者, 跨性别者, 酷儿, 性取向 *Anti-LGBTQ, Heterosexuality, Homosexuality, Lesbian, Gay, Bisexual, Transgender, Queer, Sexual Orientation*

Table D1: Topic and keywords of crawled data.

text on the Chinese web platforms, such as English and emoji (Li et al., 2020). Like profanity "ni哥" (meaning "*ni brother*"), which has the same pronunciation as "*n*gger*".

Borrowed Word. Certain toxic cultural connotations pervade some phonetic foreign words (Shi, 2010). Therefore, background information is required to clarify the actual semantics of these terms. An example is "凯勒奇", a reference to the anti-Semitic *Kalergi Program*, which is used as an inflammatory term.

D Supplement of Data Description

D.1 Data Source Information

According to their monthly financial report[10][11], *Zhihu* and *Tieba* have approximately 100 million and 40 million monthly active users, respectively. The samples in the dataset come from 5,385 users. And the samples are extracted from June 2021 to November 2022.

D.2 Keywords of Platforms

Table D1 lists the keywords of the crawled posts for diverse topics. In the process of data annotation, we note that the distribution of toxic subtype has a significant difference in the samples from the two platforms, as shown in Table D2. From the statistics, we can see that more than half of the toxic samples on *Zhihu* have subtle expressions, including implicit hate and reporting. In contrast, users from *Tieba* utilize more direct attacks to insult others, containing general offensive language and explicit hate speech. This inspires us to adapt the methodology appropriately to detect toxic language from different platforms in future work.

D.3 Samples *w* or *w/o* insults

In this section, we statistic the samples containing insults to further demonstrate the effectiveness of the two-step annotation procedure. To restore the process, we count the samples based on the profanity list at the end of the pseudo-annotation phase and the final insult lexicon, respectively. The result is shown in Table D3.

In the first stage, 3,474 samples are pseudo-labeled as toxic, and 91.1% of them are indeed toxic, representing 49% of all toxic samples. This reflects the fact that pseudo-annotation can significantly reduce the annotation burden. Afterwards, some low-frequency swear words and implicit in-

[10] https://ir.zhihu.com/Quarterly-Results
[11] https://ir.baidu.com/financial-reports

Platform	N-Tox.	Tox.	Toxic Category					Total	Avg. L
			Off.	Hate	H-exp.	H-imp.	H-rep.		
Zhihu	3,094	3,187	270	2,917	1,088	1,055	774	6,281	41.75
Tieba	2,456	3,274	546	2,728	1,649	940	139	5,730	34.99
Total	5,550	6,461	816	5,645	2,737	1,995	913	12,011	38.37

Table D2: Statistics of different platforms in ToxiCN.

Lexicon	Label	w/ Insult	w/o Insult
Pseudo	**Tox.**	3,166	3,295
	N-Tox.	308	5,242
	Total	3,474	8,537
Annotation	**Tox.**	4,331	2,130
	N-Tox.	1,162	4,388
	Total	5,439	6,518

Table D3: Statistics of samples with ("*w/*") or without ("*w/o*") any insults, where "*Pseudo*" and "*Annotation*" means the insults are from the profanity list in the pseudo labeling and the final lexicon respectively.

Num of Group Labels (n)	Size	%
1	4,802	85.07
2	788	13.96
≥ 3	54	0.96

Table D4: Statistics of hate speech against single and multi-class attacked groups. *Size* denotes the number of samples attacking n groups, and % is the percentage of matched samples of the total number of hate speech.

sults are added to the lexicon during the main manual labeling. 1,162 instances with insults are ultimately labeled as non-toxic and 2,130 without insults are toxic, showing the necessity of manual inspection. These samples are more difficult to be identified than the cases that can be filtered directly using an insult lexicon, and need to be focused on in future work. However, even so, the comments containing insults are more likely to be toxic, illustrating the significance of lexical knowledge for toxic language detection.

D.4 Samples of Attack Multi-class Group

Here we calculate the proportion of utterances attacking multi-class groups. As the result shown in Table D4, there are about 15% of the samples containing attacks and discrimination against multiple groups in ToxiCN.

D.5 Statistics of Sub-varieties of Chinese

Simplified Chinese accounts for 99.7% of the crawled data on both platforms. This reflects the reality of Chinese web platforms because Simplified Chinese is the official language of China. And we do not do additional sampling for data imbalance.

D.6 Samples of attacks on Disabilities

We note in the study of other languages, such as English, "disability" has received much attention (Davidson et al., 2017; Founta et al., 2018; Pérez-Almendros et al., 2020). However, Chinese online platforms themselves can filter out posts attacking disabled people. Therefore, few comments can be retrieved and collected. This is why existing Chinese datasets don't contain samples attacking "disability" (Deng et al., 2022; Jiang et al., 2022; Zhou et al., 2022). Meanwhile, we found some toxic samples contain "disability" related slurs, like "foolish", used to attack others. Whether these samples are attacks against "disability" is worth discussing.

In addition, it also reflects that the taxonomy (Monitor Toxic Frame), and thus all the current resources, are dependent on the filtering system of the online platforms from which the data were crawled. In the future, we will use natural language generation techniques and human adversarial attacks to complement Chinese toxic language that attacks specific groups, referenced by (Hartvigsen et al., 2022; Dinan et al., 2019).

E Experimental Details

The details of the hyperparameters are listed in Table E1. During the phase of the experiment, we note that the hyperparameters of BERT and RoBERTa with the best performance are basically the same.

Hyperparameters	BiLSTM	BERT/RoBERTa
epochs	20	5
batch size	64	32
learning rate	1e-3	1e-5
padding size	100	80
dropout rate	0.5	0.5
λ	0.5	0.01

Table E1: The hyperparameters of the experiment.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In the section on "limitations".

☑ A2. Did you discuss any potential risks of your work?
In the section on "limitations".

☑ A3. Do the abstract and introduction summarize the paper's main claims?
In the section on "abstract" and "introduction".

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

In all the sections.

☑ B1. Did you cite the creators of artifacts you used?
In all the sections.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
In the section on "Ethics Statement"

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
In the section on "Ethics Statement"

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
In the section on "3.2 Data Collection and Filtering", "Ethics Statement"

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
In the section on "3.2 Data Collection and Filtering".

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
In the section on "3.4 Data Description", "6.2 Implementation" and "Appendix B".

C ☑ Did you run computational experiments?

In the section on "6 Experiments"

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
In the section on "6.2 Implementation" and "Appendix D".

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
In the section on "6.2 Implementation".

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
In the section on "6 Experiments"

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
In the section on "3.3.3 Annotation Procedure", "6 Experiments"

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
In the section on "3.3.2 Mitigating Bias"

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
In the section on "Ethics Statement"

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
In the section on "3.3.2 Mitigating Bias"

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
In the section on "Ethics Statement"

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
In the section on "Ethics Statement"

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
In the section on "3.3.2 Mitigating Bias"