UIRISC at SemEval-2023 Task 10: Explainable Detection of Online Sexism by Ensembling Fine-tuning Language Models

Tianyun Zhong, Runhui Song, Xunyuan Liu, Juelin Wang, Boya Wang, and Binyang Li*
Lab of Intelligent Social Computing
University of International Relations, Beijing, China
{tyzhong,rhsong,xyliu,jlwang,bywang,byli}@uir.edu.cn

Abstract

Under the umbrella of anonymous social networks, many women have suffered from abuse, discrimination, and other sexist expressions online. However, exsiting methods based on keyword filtering and matching performed poorly on online sexism detection, which lacked the capability to identify implicit stereotypes and discrimination. Therefore, this paper proposes a System of Ensembling Fine-tuning Models (SEFM) at SemEval-2023 Task 10: Explainable Detection of Online Sexism. We firstly use four task-adaptive pre-trained language models to flag all texts. Secondly, we alleviate the data imbalance from two perspectives: oversampling the labelled data and adjusting the loss function. Thirdly, we add indicators and feedback modules to enhance the overall performance. Our system attained macro F1 scores of 0.8538, 0.6619, and 0.4641 for Subtask A, B, and C, respectively. Our system exhibited strong performance across multiple tasks, with particularly noteworthy performance in Subtask B. Comparison experiments and ablation studies demonstrate the effectiveness of our system.

1 Introduction

Sexism refers to prejudice, stereotyping, or discrimination based on one's gender or sex, typically against women(Wikipedia contributors, 2023). Sexist expressions cause gender stereotypes and discrimination, such as "whxxe" or "Husbands. Kill your piece of sxxt commie wives[1]". Especially with the widespread and fast propagation of social media, the negative impact of gender discrimination has been further exacerbated. Online sexist texts can not only affect the user experience and community environment but also lead to offline violence, persecution even crimes, which may cause much harm to real society. It is essential to eliminate sexist expressions and build a harmonious community.

For this reason, many previous studies have focused on capturing offensive posts and comments(Chen et al., 2012; Davidson et al., 2017). These methods usually filtered texts by keywords matching e.g. lexicon-based models. However, the increasing number of active users made it ineffective of lexicon based methods. Moreover, many expressions do not contain indicative words e.g "bixxh", but they still convey strong sexism and prejudice as well, such as "I always cancel as soon as the driver accepts my ride is a female. Then immediately rebook[1]". They both in turn affect the performance of sexism detection.

In this paper, we propose a computational system named System of Ensembling Fine-tuning Models (SEFM) for Semeval-2023 Task 10: Explainable Detection of Online Sexism (EDOS)(Kirk et al., 2023). SEFM consists of three modules: Data Preprocessing, Sexsim Detection, and Ensembling. In the data preprocessing module, we extend the original dataset by Easy Data Augmentation (EDA). The sexism detection model includes three improvements to enhance the model effect: Sexism Indicator for subtask A (SIA), Feedback for subtask B (FB), and Fine-graind Indicator for subtask C (FIC). The codes will be open sourced[2].

The rest of the paper is organized as follows: Section 2 gives a brief literature survey. Section 3 introduces our system. Section 4 describes the experimental setups, while Section 5 demonstrates the results and makes the analysis. Finally, we reach the conclusions in Section 6.

2 Background

Semeval-2023 holds Task 10: Explainable Detection of Online Sexism, which contains three sub-

*Corresponding author

[1]This sentence was selected from the website by Semeval-2023 Task 10's organizers.

[2]https://github.com/tianyumyum/UIRISC-SemEval2023Task10

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 2082–2090
July 13-14, 2023 ©2023 Association for Computational Linguistics

tasks to flag what is sexist content and explain why it is sexist, which aims to approach explainable sexism detection via the granularity of classification labels[3].

2.1 Task Introduction

As shown in Figure 1, EDOS is aimed at sexism detection that is more accurate as well as explainable, with fine-grained classifications for sexist content from Gab and Reddit.

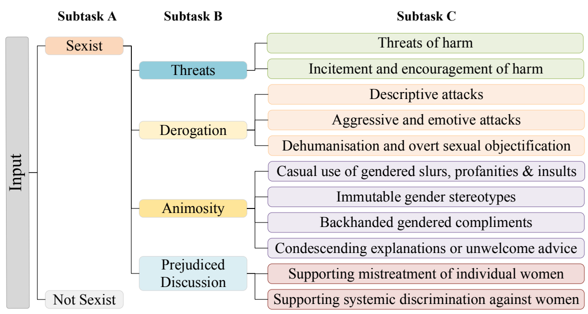

Figure 1: Task overview.

- SUBTASK A - Binary Sexism Detection: the task requires a two-class classification that requires the system to predict whether a post is sexist or not based on the content.
- SUBTASK B - Category of Sexism: the task requires a four-category classification according to the degree of sexism on sexist posts, where the system must predict one of four categories: (1) threats, (2) derogation, (3) animosity, (4) prejudiced discussions.
- SUBTASK C - Fine-grained Vector of Sexism: for posts which are sexist,this task requires an 11-class classification, and the system must predict one of 11 fine-grained vectors, all based on the Task B classification.

2.2 Related Work

Many studies have focused on automated methods to effectively detect hate speech detection and sexism classification. Waseem and Hovy (2016) explored the role of extra-linguistic features with character n-grams in classifying tweets as racism, sexism, or neither. Badjatiya et al. (2017) tried various deep-learning approaches for the same three-way classification. Zhang and Luo (2018) explored skipped CNN and a combination of CNN and GRU for hate speech detection. They presented the first attempt to categorize comments involving any type(s) of sexism in a multi-label way. Zia et al. (2022) employed pseudo-label fine-tuning of Transformer Language Models to detec automatic hate speech. Samory et al. (2021) applied psychological scales to detect different dimensions of sexism.

More recently, pre-trained language models(PLM) such as BERT(Devlin et al., 2018), ERNIE(Zhang et al., 2019), and GPT-3(Brown et al., 2020), have set the new state-of-the-art in hate speech detection and sexism classification tasks. It has also become a consensus to fine-tune large-scale PTMs for specific AI tasks, rather than learning models from scratch(Qiu et al., 2020). In order to adapt language models to domains and tasks, Gururangan et al. (2020) pre-trained different domain unlabeled data into RoBERTa model, whose performance exceeds RoBERTa in all tasks. So far, various efforts have been made to explore large-scale PTMs in text classification tasks(Tian et al., 2020; Rezaeinia et al., 2019).

3 System Overview

The framework of our system is shown in Figure 2, and the detailed description for each part is presented as follows.

3.1 Data Preprocessing

Considering the colloquial and non-standard characteristics of text originating from social media, we pre-processed the data according to the following steps.

- **Removal of meaningless words**

  We identified certain words in the dataset that carried no actual meaning, such as "[URL]" and "[USER]", and proceeded to eliminate them. We carried out further experiments by eliminating stopwords, but observing a slight degradation in the performance.

- **Emoji interpretion**

  We utilized the emoji library[4] to retain the emotional content conveyed by emojis. By incorporating this information, we were able to enhance the accuracy and nuance of our insights into the emotional content of the text data.

[3] https://codalab.lisn.upsaclay.fr/competitions/7124#

[4] https://pypi.org/project/emoji/

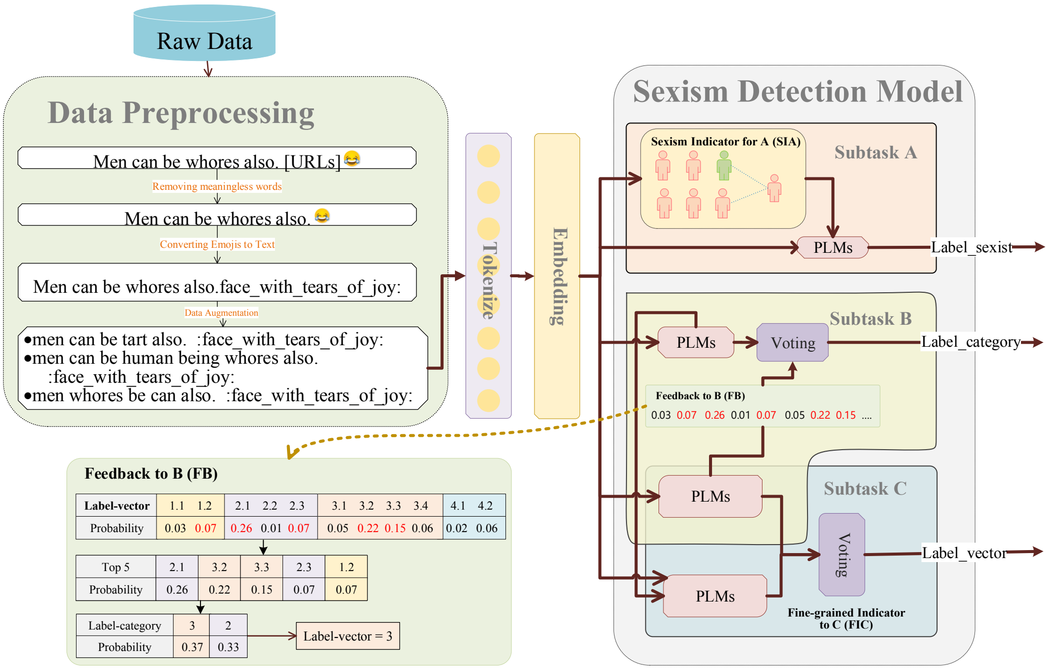

Figure 2: Overview of SEFM. Firstly, the data was pre-processed according to the above steps. Then, the model section module consists of three key components: SIA learns the voting details from different annotators, FB and FIC mutually reinforce each other to enhance the performance of subtask B and C. Finally, we make full use of the augmented data to optimize the model output through a voting mechanism.

- **Data augmentation**

  Class distribution of the sample is unbalanced, which may lead to a significant bias in our detection model. To address this issue, we employed an oversampling approach. To prevent overfitting, we added slight random perturbations to each data expansion(Wei and Zou, 2019) [5], including (1) synonym replacement, (2) random insertion of synonyms, (3) random exchange of word order, and (4) random deletion of words.

3.2 Sexism Detection

3.2.1 Task-adaptive language model construction

To capture semantic information beyond the limitations of keyword matching, such as with W2V, our system employs deep learning networks. Fine-tuning different PLMs with varying hyperparameters and training strategies has gained popularity for its capability to extract and integrate contextual information, thereby enabling more precise language processing (Song et al., 2021; Yuan et al., 2022).

Considering that direct exposure to more data without considering domain relevance is detrimental to the final task performance(Gururangan et al., 2020), we adopt task-adaptive pre-training (TAP) to boost our results. We use vast amounts of unlabelled data into Masked LM and next sentence prediction. Pre-training is continued with more task-relevant unlabeled data to output an adaptive pre-trained model for this task.

3.2.2 Sexism indicaror for subtask A

The EDOS task publisher provided public with the edos_labelled_individual_annotations.csv[6]. Disagreements may have arisen among the three annotators during the annotation process of the training set, potentially compromising the quality of the data. We observed that although each text was annotated by three annotators, only two of them flagged consistently for the some text.

Base on this dataset, we first labeled each text as either Gold or Silver. If the three annotators agreed, the data was considered Gold, and if there was disagreement, the data was considered Silver. Table 1 shows an example.

We designed an indicator to assign different weights to Gold and Silver during model training.

[5]https://github.com/jasonwei20/eda_nlp

[6]https://github.com/rewire-online/edos/ tree/main/data

Rewire_id	Text	Annotators	Label	Confidence
sexism2022_english-10016	Pls she makes me want to vomit, I do not want to see her face or give her any attention. no offense, DG.	1 2 9	not sexist sexist not sexist	SILVER
sexism2022_english-1000	Just imagine if she came in wearing a hijab.	1 3 17	not sexist not sexist not sexist	GOLD

Table 1: Example of SIA procedure.

We then used these labeled texts as new input for training the pre-trained model, and compared the results with our previous approach.

3.2.3 FB and FIC module for subtask B and C

We noticed that subtask B and C shared the same training dataset and had identical task release and deadline dates. Therefore, besides improving the model architecture, we introduced new modules for subtask B and C respectively. For subtask B, we added a new module called FB (Feedback to B) to receive the results from subtask C detection. The entire process involved preprocessing the text, using task-adaptive pre-trained models to derive the results, and receiving feedback from subtask C. Finally, majority voting was performed on the output of the two branches to obtain the final output.

Similarly, for subtask C, we introduced the FIC (Fine-graind Indicator for subtask C) to receive the detection results from subtask B. One branch of subtask C involved an 11-class classification task, while the other branch refined the classification based on the subtask B results for explainability.

3.3 Loss Function

Loss function is used to evaluate the extent to which the predicted and true values of the model are not the same. For different models and different tasks, the choice of loss function has a great impact on the performance of the model. In this task, the focal loss function is used to better alleviate the problem of unbalanced number of sample categories.

$$\begin{aligned} \text{BCE loss}(o,t) = -1/n \sum_i (t[i] * \log(o[i]) \\ +(1-t[i]) * \log(1-o[i])) \end{aligned} \tag{1}$$

As shown in equation 1, we use balance factor to alleviate data imbalance in Balance Cross Entropy loss(BCE loss).

$$\mathrm{FL}\left(p_{\mathrm{t}}\right)=-\alpha_{\mathrm{t}}\left(1-p_{\mathrm{t}}\right)^\gamma \log \left(p_{\mathrm{t}}\right) \tag{2}$$

Focal loss is specially designed for the one-stage detection algorithm, which reduces the loss weight of easy-to-distinguish negative examples. It increases the dynamic adjustment factor based on BCE loss to achieve the effect of difficult sample mining. We make the model more focused on hard-to-learn samples by setting γ value as 2 in the equation 2, thus the network will not be biased by too many negative examples.

4 Experiments

4.1 Dataset

SemEval-2023 Task 10 dataset(Kirk et al., 2023) comprises 14,000 annotated instances, yet suffers from imbalanced data distribution among the categories for all three subtasks. Subtask C, in particular, exhibits a significant class imbalance, with the label "3.4 condescending explanations or unwelcome advice" having only 54 instances in the training dataset. This could hinder the model's ability to learn sufficient features for accurate predictions. Figure 3 illustrates the distribution of instances for each label in the training dataset.

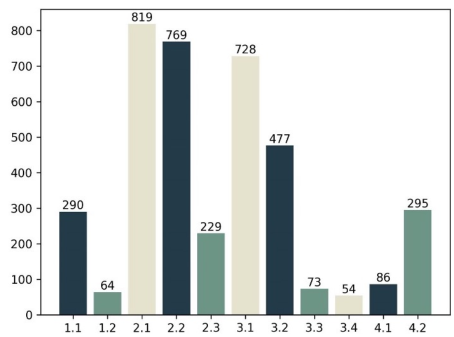

Figure 3: Data distribution of subtask C.

4.2 Experiment Setup

We utilized the PyTorch library (Paszke et al., 2019) and the HuggingFace library (Wolf et al., 2020) our models and trained and tested them on Kaggle GPUs. We split the entire dataset into a 90%

	Subtask A			Subtask B			Subtask C		
	P	R	F1	P	R	F1	P	R	F1
BERT	0.7655	0.8056	0.7850	0.6113	0.6122	0.6117	0.3769	0.4635	0.4157
ALBERT	0.7724	0.8434	0.8064	0.6394	0.6152	0.6270	0.4220	0.3971	0.4092
RoBERTa	0.8006	0.7298	0.7636	0.6113	0.6436	0.6270	0.4131	0.4949	0.4503
ERNIE2.0	0.7934	0.7846	0.7890	0.6040	0.7222	0.6578	0.3997	0.4192	0.4092
BERT+SEFM	0.7964	0.8137	0.8049	0.6304	0.6878	0.6578	0.3992	0.5106	0.4481
ALBERT+SEFM	0.7757	0.8573	0.8144	0.5900	0.7252	0.6507	0.3760	0.4648	0.4157
RoBERTa+SEFM	0.8185	0.8718	0.8443	0.5804	0.7507	0.6547	0.4239	0.4803	0.4503
ERNIE2.0+SEFM	0.8266	0.7871	0.8064	0.6314	0.6717	0.6509	0.3848	0.5364	0.4481
Ours	**0.8341**	0.8745	**0.8538**	**0.6635**	0.6603	**0.6619**	**0.4275**	0.5076	**0.4641**
(Das et al., 2022)	0.8200	0.8000	0.8100	0.5900	0.5500	0.5700	0.3800	0.3700	0.3700

Table 2: Comparison between PLMs and PLMs+STFM in Subtasks A, B, and C.

Subtask	P	R	Macro F1	Rank
A	0.8536	0.8540	0.8538	19/84
B	0.6603	0.6635	0.6619	12/69
C	0.4938	0.4533	0.4641	20/63

Table 3: Results of subtask A, B, and C.

training set and a 10% development set. We used the Adam optimizer with a learning rate of 1e-3 and a weight decay coefficient of 1e-6. The batch size was set to 16, and the models were trained for 2 epochs. We adopted accuracy, precision, recall, and macro f1 score as the evaluation metrics.

4.3 Baselines

To evaluate the performance of our system, we applied it to the following methods and compared the results before and after the application.

- **BERT** (Devlin et al., 2018) utilized masked language model to generate deep bidirectional linguistic representations and achieved SOTA performance in various downstream tasks.
- **ALBERT** (Lan et al., 2019) proposed an improvement on BERT by integrating two techniques, which contributes to a smaller number of parameters and faster training speed.
- **RoBERTa** (Liu et al., 2019) employs a larger number of model parameters, more training data, and a larger batch size.
- **ERNIE2.0** (Sun et al., 2019) was able to extract valuable information, including vocabulary, syntactic, and semantic representations from the training corpus.

4.4 Ensemble

For the final output, we apply a majority voting to ensemble several models (Ganaie et al., 2022). Given that we employ data augmentation during data preprocessing, a single "rewire_id" can correspond to multiple similar texts after model detection. Majority voting aggregates the predictions of different outputs and determines the final label.

5 Results and Analysis

We submitted the scores predicted by the ensemble method introduced above. The official ranking is presented in Table 3. In subtask B, we ranked 12th, which verifies the validity of our system.

5.1 Comparison Experiments

5.1.1 Comparison on different models

Table 2 presents the results of online sexism detection. In our experiments, we evaluate the performance of our system by applying it to the following methods and comparing the Macro-F1 results before and after the application. We first test the four baselines on three subtasks, and then use the best settings with our system on four pre-trained models for comparison.

5.1.2 Comparison on different loss functions

To better evaluate the impact of Focal loss in the system, we experimented with three different loss functions. In order to better alleviate the problem of unbalanced number of sample categories, we used the focal loss function.

Among the three loss functions, BCEloss weighted loss alleviates the problem of number balance among samples and performs better than Cross Entropy loss (CEloss). Focal loss not only alleviates the problem of sample imbalance, but also

		A			B		
		P	R	F1	P	R	F1
Ours	w/ Cross Entropy	0.7452	0.8014	0.7723	0.5215	0.5528	0.5367
	w/ Balanced Cross Entropy	0.8245	0.8636	0.8436	0.6424	0.6440	0.6432
	w/ Focal Loss	0.8341	0.8613	**0.8475**	0.6603	0.6635	**0.6619**

Table 4: Comparison of three different loss functions.

	EDA	DevEDA	A			B		
			P	R	F1	P	R	F1
Ours	✓		0.8154	0.7939	0.8045	0.6112	0.6330	0.6219
		✓	0.8168	0.8488	0.8325	0.6175	0.6480	0.6324
	✓	✓	0.8241	0.8645	**0.8438**	0.6084	0.6793	**0.6419**

Table 5: Comparison of data argumentation method.

	Dev			Test		
	P	R	F1	P	R	F1
Ours	0.8670	0.8045	**0.8346**	0.8132	0.8586	**0.8322**
- SIA	0.8266	0.7907	0.8063	0.8086	0.8409	0.8228

Table 6: Validation of sexism indicator to subtask A.

	B			C		
	P	R	F1	P	R	F1
Ours	0.6603	0.6635	**0.6619**	0.4275	**0.5076**	**0.4641**
- FB, FIC	**0.6473**	0.6455	0.6464	0.4146	0.4345	0.4243
- FB	0.6367	0.6691	0.6525	0.4363	0.4699	0.4525
- FIC	0.6324	**0.6845**	0.6574	**0.4525**	0.4649	0.4586

Table 7: Validation of improvement to subtask B and C.

incorporates detection difficulty into the formula and performs the best among the three. The results are shown in Table 4.

5.1.3 Improvement by data augmentation

Data augmentation is a useful technique for increasing a model's generalization capabilities and can also address many other challenges and problems, from overcoming a limited amount of training data to regularizing the objective (Bayer et al., 2021). In this task, data augmentation differs from the oversampling operation of directly copying the data by using insertion, deletion, and replacement operations on the sample data to avoid overfitting. The results are shown in Table 5.

5.2 Ablation Studies

To demonstrate the effectiveness of the Indicator and Feedback components, we also conducted ablation studies with the following experiments:

- SIA: removing the Indicator to A module, the train data is the official version without weighing the confidence of label.

- FB: removing the Feedback from B module, subtask B is directly divided into four categories, without ensemble the results from Task C.

- FIC: removing the Indicator to C module, subtask C selects from 11 vectors with the highest probability after the Softmax layer, without fusion of subtask B.

The ablation experiments for subtask A are shown in Table 6, and the ablation experiments for subtasks B and C are shown in Table 7.

5.3 Comparison on Ensemble Combination

To better explore the results of ensemble, we validated the four pre-trained models with different combinations. As shown in Table 8, the outputs after majority voting don't show obvious improvements.

Since the pre-trained models are all BERTs or variants of BERTs with less complementarity between them, it is more difficult to achieve the improvement of results directly through ensembles.

5.4 Error Analysis

5.4.1 Diversity Analysis of Model Results

We analyzed our experimental results and found that ensembling exclusively BERT variants did not offer significant improvement over individual best-performing variants. However, as Kuncheva and Whitaker (2003) point out, diversity among models is a crucial factor in explaining the performance gains achieved by ensembles. The idea for our measure came from the work of Hansen and Salamon (1990). We verified the relationship between diver-

Ensemble Models	Initial Macro F1	Macro F1
AlBERT+ERNIE	0.8683;0.8387	0.8412
RoBERTa+ERNIE	0.801;0.8387	0.8212
BERT+ERNIE	0.8538;0.8387	0.8453
RoBERTa+ALBERT	0.8234;0.8683	0.8510
BERT+Albert	0.8538;0.8683	0.8279
BERT+RoBERTa	0.8538;0.801	0.8316
BERT+RoBERTa+albert	0.8538;0.801;0.8683	0.8542
BERT+RoBERTa+ERNIE	0.8538;0.801;0.8387	0.8422
BERT+ALBERT+ERNIE	0.8538;0.8683;0.8387	0.8657
RoBERTa+ALBERT+ERNIE	0.801;0.8683;0.8387	0.8562
BERT+RoBERTa+ALBERT+ERNIE	0.8538;0.801;0.8683;0.8387	0.8638

Table 8: Combinations of the four pre-trained models.

sity and correctness using a measure of diversity based on the distribution of difficulty.

Let $\mathcal{D} = \{D_1, \ldots, D_L\}$ represent a set of models and $\mathcal{P} = \{P_1, \ldots, P_L\}$ denote the set of accuracy of models in $\mathcal{D}$. We define a discrete random variable X that takes values in $0, 1/L, ..., 1$ and indicates the proportion of classifiers in $\mathcal{D}$ that correctly classify a given input $\mathbf{x}$ inferred from texts.The experimental data for the error analysis were 2,000 validation set data in Subtask A and 486 data in Subtask B(Kirk et al., 2023).

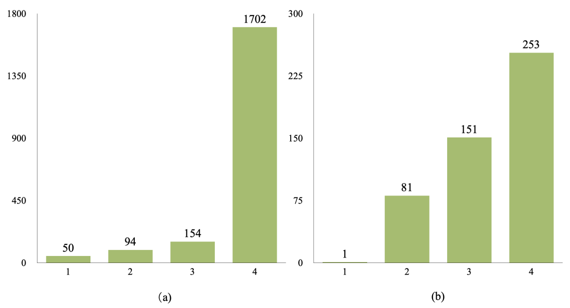

Figure 4: The histograms in both graphs depict the number of texts that were labeled the same result by 'i' models. The x-axis represents the number of models showing the same results(i.e., i). The number of outputs generated in D for Subtask A and Subtask B are illustrated in (a) and (b), respectively.

As shown in Figure 4, Out of the 2,000 data points for Task A, 1,702 data points showed identical results across the four models, while just under 15% of the data points showed some level of divergence across the four models. For Task B, out of the 486 data points, 253 data points yielded identical results across the four models and 151 data points yielded identical results across the three models. These results suggest that the four models produced very similar outputs.

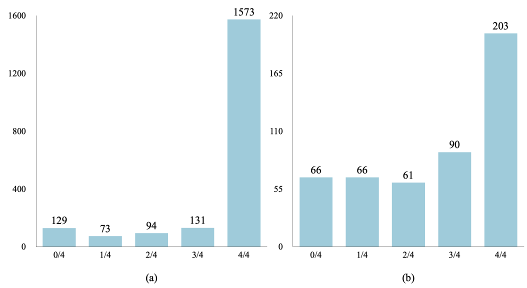

Figure 5: Patterns of "difficulty" for three classifier teams with $L = 4$. The x-axis is "proportion correct", i.e., i/L.. The numbers of texts which are correctly labeled by i of the L models for Subtask A and Subtask B are illustrated in (a) and (b), respectively.

Figure 5 displays X's histogram for $L = 4$. High consistency of the models is observed when data is distributed at the ends.In Subtask A and Subtask B, our experiments showed that less than 14.9% of the data varied between the models in $\mathcal{D}$. The difficulty θ is defined as $\mathrm{Var}(X)$, scaled into [0,1], where higher values indicate poor classifier team performance. $\theta_a = 0.7531$ and $\theta_b = 0.1852$ for Subtask A and Subtask B, respectively. These results suggest a lack of diversity in the models' outputs in $\mathcal{D}$, which may limit ensemble performance. Only 11 texts were corrected by the ensemble in Task A compared to the individual Roberta model results.

5.4.2 Case Studies

We analyzed the limitations of model predictions in $\mathcal{D}$ by examining texts that four models couldn't classify. Examples in Table 9 show the models

Rewire_id	Text	Pred_sexist	Label_sexist
sexism2022_english-11494	U thawt 'chu had a fweind biyatch	not sexist	sexist
sexism2022_english-17521	Maybe gay marriage. But with a woman? You want me to have to be partnered for all my life with a demon? No.	not sexist	sexist
sexism2022_english-18794	"Some are, some aren't. Some thin women will date bigger guys."	not sexist	sexist
sexism2022_english-18165	Good point, i went on a tinder date once and the girl didn't mind not talking all the time which i found incredibly refreshing, too bad i wasn't physically attracted.	sexist	not sexist

Table 9: Case studies.

struggle with typing errors, slang, spoken words, metaphors, and lengthy sentences. Colloquial expressions and metaphors in dataset texts make it hard for models to learn features. Ambiguous sentences in the dataset also challenge categorization.

6 Conclusions

This paper presents our system implementation submitted to the SemEval-2023 Task 10: Explanable Detection of Online Sexism. We propose an Ensemble Framework named System of Ensembling Fine-tuning Models (SEFM) that enhances system performance by pre-processing data, training Task-adaptive PLMs, and adding Indicator and FB modules. In the future, we plan to utilize the dataset further and improve our system by introducing the prompt module and fusion label vector to enhance the performance of online sexism detection.

Acknowledgements

We would like to thank Zhengyi Zhao, who offered us invaluable advice on improving the experiments. Additionally, we extend our gratitude to Yaojie Lu for his insightful comments and suggestions, which greatly contributed to the completion of this paper.

This project was partially supported by National Natural Science Foundation of China (Grant number: 61976066), Beijing Natural Science Foundation (Grant number: 4212031), the Fundamental Research Fund for the Central Universities (Grant numbers: 3262023T19), and Research Funds for NSD Construction, University of International Relations (Grant numbers: 2021GA07).

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, WWW '17 Companion, page 759–760, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2021. A survey on data augmentation for text classification. *CoRR*, abs/2107.03158.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80.

Mithun Das, Somnath Banerjee, and Animesh Mukherjee. 2022. Data bootstrapping approaches to improve low resource abusive language detection for indic languages. In *Proceedings of the 33rd ACM Conference on Hypertext and Social Media*, pages 32–42.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mudasir A Ganaie, Minghui Hu, AK Malik, M Tanveer, and PN Suganthan. 2022. Ensemble deep learning:

A review. *Engineering Applications of Artificial Intelligence*, 115:105151.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Lars Kai Hansen and Peter Salamon. 1990. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001.

Hannah Rose Kirk, Wenjie Yin, Bertie Vidgen, and Paul Röttger. 2023. SemEval-2023 Task 10: Explainable Detection of Online Sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation*, Toronto, Canada. Association for Computational Linguistics.

Ludmila I Kuncheva and Christopher J Whitaker. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

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.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

Seyed Mahdi Rezaeinia, Rouhollah Rahmani, Ali Ghodsi, and Hadi Veisi. 2019. Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems with Applications*, 117:139–147.

Mattia Samory, Indira Sen, Julian Kohne, Fabian Flöck, and Claudia Wagner. 2021. "call me sexist, but...": Revisiting sexism detection using psychological scales and adversarial samples. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15, pages 573–584.

Bingyan Song, Chunguang Pan, Shengguang Wang, and Zhipeng Luo. 2021. Deepblueai at semeval-2021 task 7: Detecting and rating humor and offense with stacking diverse language model-based methods. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1130–1134.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2019. Ernie 2.0: A continual pre-training framework for language understanding. *arXiv preprint arXiv:1907.12412*.

Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. 2020. Skep: Sentiment knowledge enhanced pre-training for sentiment analysis. *arXiv preprint arXiv:2005.05635*.

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

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Wikipedia contributors. 2023. Sexism — Wikipedia, the free encyclopedia. [Online; accessed 23-April-2023].

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Mengfei Yuan, Zhou Mengyuan, Lianxin Jiang, Yang Mo, and Xiaofeng Shi. 2022. stce at semeval-2022 task 6: Sarcasm detection in english tweets. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 820–826.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. *CoRR*, abs/1905.07129.

Ziqi Zhang and Lei Luo. 2018. Hate speech detection: A solved problem? the challenging case of long tail on twitter. *CoRR*, abs/1803.03662.

Haris Bin Zia, Ignacio Castro, Arkaitz Zubiaga, and Gareth Tyson. 2022. Improving zero-shot cross-lingual hate speech detection with pseudo-label fine-tuning of transformer language models. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 16, pages 1435–1439.