EVALUATING GPT-OSS-20B MODEL FOR HATE SPEECH DETECTION: ADVANCES IN PARAMETER-EFFICIENT ADAPTATION

Quang Hong Le[1,*], Tan Khoa Ly Huynh[1], Thanh Le[2]

[1] Ho Chi Minh City University of Foreign Languages - Information Technology, Ho Chi Minh City, Vietnam

[2] Faculty of Information Technology, HUTECH University, Ho Chi Minh City, Vietnam

quanglh@huflit.edu.vn, huynhlytankhoa@gmail.com, lthanh25nct@hutech.edu.vn

ABSTRACT— The problem of detecting hate speech is methodologically difficult because of the ambiguity of annotations, the class imbalance issue, and the blurry border between offensive and hateful text examples. This paper evaluates the efficiency of using a 20-billion-parameter large language model, adapted by a parameter-efficient method to perform hate speech classification in three categories. Specifically, we apply the following techniques to our data: (i) annotation consolidation; (ii) balanced sampling to reduce minority classes' sparseness; (iii) instruction templates along with consensus-based metadata augmentation to provide better prediction stability in borderline cases. We compare the performance of this approach to the transformer encoder baseline and the prompted large language model baselines. Our results show that this approach allows us to reach the macro F1 score of 80.66% and accuracy of 83.37%. Moreover, the performance gain in the Hate Speech class is significant compared to all the baselines considered. Furthermore, a resource utilization analysis demonstrates that this adaptation process is highly computationally efficient.

Keywords— Hate speech detection; Large language models; QLoRA fine-tuning; Instruction tuning.

I. INTRODUCTION

The Internet has emerged as the dominant channel through which people engage in conversations; nevertheless, its rapid expansion has been paralleled by an increase in the prevalence of abusive and hateful language.

Consequently, identifying and classifying offensive language is crucial for protecting the interests of individuals and creating scalable systems for content moderation. Although considerable progress has been made within the field of natural language processing, categorizing text into Normal, Offensive, and Hate Speech has proven difficult owing to similarities in word choice, style, and underlying intent among the three types of communication [1], [2].

In order to understand the intricacies involved, HateXplain was created to incorporate three different annotations, rationale spans, and target-group information for each example. It highlights the difficulty in conducting fine-grained classification, where there is a lack of consensus among annotators, ambiguity in determining whether something falls within Offensive or Hate Speech, and a scarcity of minority groups. As a result, traditional cross-entropy loss methods may be unstable and biased towards majority labels when training models. Recent experiments confirm that LLMs are able to recognize more detailed semantics but require specialized training approaches due to differences in performance across classes [3].

Prompted by these challenges, this paper explores the fine-tuning of GPT-OSS-20B through a QLoRA-based fine-tuning process specifically tailored towards the features of HateXplain. The methodology involves three aspects: (i) the use of a rule-based approach for resolving differences between annotators; (ii) the inclusion of metadata into instructions in the form of targeting groups and reasoning labels; and (iii) balancing using both weighted loss and weighted sampling methods. These factors are intended to boost the performance of the model regarding its ability to capture annotator biases and make fine distinctions between linguistic patterns while ensuring computational efficiency for this large model.

The main contributions of this paper are as follows:

- We introduce a fine-tuning framework for GPT-OSS-20B that integrates annotator-aware label derivation, metadata-enhanced instruction prompts and a dual class-balancing strategy tailored to the characteristics of the HateXplain corpus.
- We provide a comprehensive evaluation of model behaviour across the Normal, Offensive and Hate Speech classes, with specific attention to systematic confusion affecting borderline Offensive cases.
- The effectiveness analysis of QLoRA adaptation on a 20-billion parameter model in restricted hardware environment is done along with an error analysis to explain the effect of ambiguity and disagreement among annotators on classification errors.

The rest of this paper will be structured as follows: In section 2, we present related works in hate speech detection, fine-grained classification of abusive language, and instruction-tuned models at scale. The methodology employed in our work is presented in section 3, which includes data pre-processing, label creation, metadata generation,

QLoRA training settings, and class balancing techniques. Section 4 outlines our experiments and their results. Section 5 highlights classification mistakes in our analysis and provides insights into our methods. Finally, the conclusions are drawn in section 6.

II. RELATED WORK

Methodologies for hate speech detection have shifted significantly in recent years. Current literature has shown that transformer models have replaced previous machine learning approaches in terms of being able to detect contextual features, sentiment analysis, and negative connotations [1]. Despite having achieved high levels of accuracy due to such innovations, studies continue to show persistent problems that occur with these models, including class imbalance, the difficulty in differentiating between offensiveness and hatred, as well as the subjectivity of annotators' opinions. All of these affect multi-class classification because categories are differentiated both semantically and contextually.

Many studies focus on corpora specifically designed for the analysis of harmful language. Among these, corpora that provide rationale or segment-specific information about harmful language have received considerable attention since they allow the evaluation of how well the models perform beyond simple statistics [5]. Studies on rationale-based annotations show that feeding models annotated spans from humans helps minimize the potential bias associated with model predictions as well as detect implicit harm [6]. Building upon this idea, some studies explore the effects of annotation rationales in the cross-lingual setting and conclude that models pretrained on English-based datasets may not generalize well without metadata or cross-lingual learning strategies [2]. Following this trend, more recent work explores the importance of annotation variations in cross-lingual and cross-domain settings [7].

However, recent studies on hate speech detection in multilingual and real-life settings expand on this viewpoint by examining the impact of diverse linguistics on transformer architectures and LLMs. Findings suggest there is significant disparity in model performance between languages and contexts even when pretrained on high-performing models [8]. Additionally, robustness-oriented literature also shows that adversarial examples can disrupt decision boundaries for offensive language classifiers, making it necessary to study the effectiveness of fine-grain evaluation datasets like HateXplain in detecting failure cases and sensitivity to lexical variations even if not used as training sets [9].

Advances in the use of LLMs for harmful content detection have expanded the scope of methodological research. The empirical performance of models ranging from 7b to 70b parameters suggests that LLMs may be superior to classical transformer encoders in zero-shot and few-shot configurations, although they are prone to instability when minority classes are underrepresented [9]. Further systematic experiments confirm that LLMs need to be prompted or fine-tuned appropriately to avoid assigning borderline classes to the majority label [10]. Parameter-efficient fine-tuning has emerged as a promising method for adapting large models to specialized data sets. Methods like LoRA and QLoRA [4] allow fine-tuning within limited hardware resources by maintaining most of the representation power of the model, and recent reviews provide empirical evidence that these methods are applicable even for models larger than a few billion parameters [11].

The references above lay the groundwork for the methodological choices made in this paper: annotation-aware label generation, enriched instruction prompts using metadata information, and balancing of classes acknowledge the general agreement that the architecture of HateXplain requires targeted adaptation rather than standard fine-tuning.

III. PROPOSED MODEL

Figure 1 illustrates the end-to-end processing pipeline of the proposed framework. After taking in the input HateXplain dataset, the framework sequentially performs data preprocessing and label harmonization, data creation from instructions, QLoRA based model tuning, fine-tuning on supervision, and finally, inference with evaluation. This entire workflow describes the process from which input datasets are fed into models after their preparation.

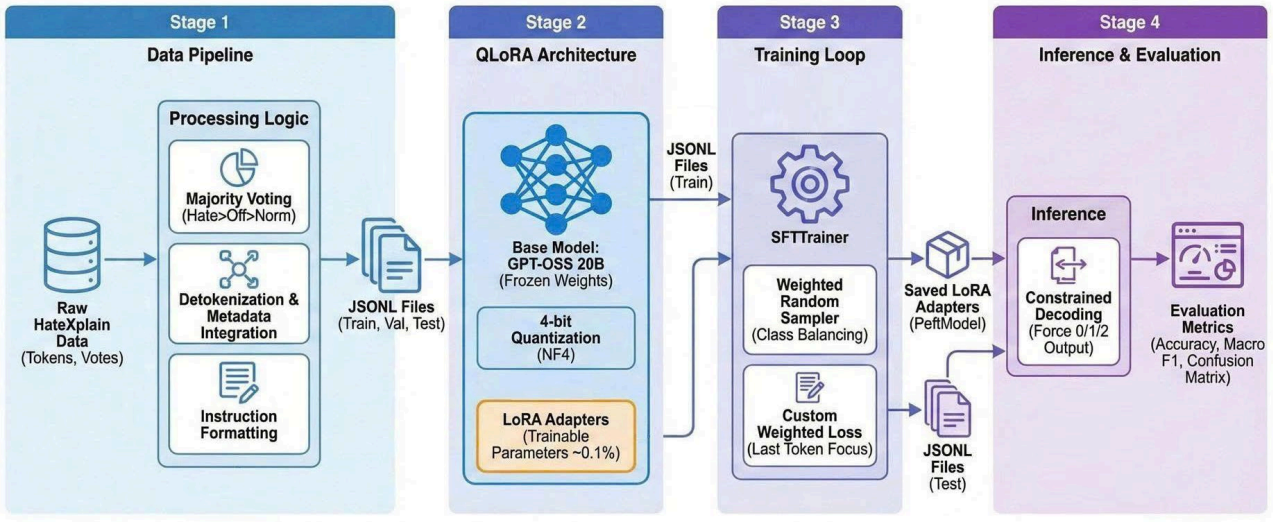

***Figure 1**. Overall workflow of the proposed method.*

A. DATA PREPROCESSING AND LABEL CONSOLIDATION

Each instance of HateXplain requires three independent labels which need to be reconciled into one golden label before fine-tuning. For this purpose, the first step involves counting the number of votes in each category. If there is a decisive winner among the categories, it is chosen as the final label. However, in the case when the annotators reach a tie situation, the order of preference based on severity comes into play. Hate Speech gets higher precedence than Offensive, which in turn gets higher precedence than Normal, and thus minimizes the chances of confusing ambiguous and offensive content with less harmful ones.

Besides the golden label, other types of context provided for each sample in the preprocessing phase include mentions of target groups mentioned in the comment itself. Votes from different annotators are kept for training samples, which indicate the level of controversy among human annotations. Furthermore, rationale tokens are kept, which help explain the decision making process of the annotators.

B. METADATA-ENRICHED INSTRUCTION CONSTRUCTION

In post-processing, each instance is converted into the instruction tuning format. In this case, the task involves serving as an adjudicator by assessing the text of the comment and relevant metadata and returning a number indicative of one out of three classes. The instruction template includes not only the comment but also the target audience(s), rationales, and voting distribution in the case of training split. Including all those aspects into the instruction provides the model with the access to human reasoning especially in the cases when the boundaries between categories become blurred.

One should pay attention to the fact that there is a considerable imbalance in class distribution in the dataset. Therefore, the weighted cross-entropy loss penalizes the rare categories with more losses according to the scheme of inverse frequency raised to a square root but only for the prediction token because in other cases, the penalty would be too large due to its multiplication through the sequence. Weighted random sampling makes the training batches closer to the desired proportions by increasing the probability of selection for Offensive and Hate Speech instances.

C. QLORA-BASED ADAPTATION AND TRAINING CONFIGURATION

Training in GPT-OSS-20B is done using QLoRA, where the backbone is initialized in 4-bit NF4 quantization, along with double quantization compression. This makes the model memory-efficient without compromising numerical stability. LoRA adapters with rank 32, $\alpha = 64$, and a dropout rate of 0.1 are added to all linear layers, and all backbone parameters are frozen during training. This allows efficient adaptation of a 20-billion-parameter model on a single high-memory GPU. Specifically, the whole model, the optimizer states, and the LoRA adapters fit comfortably within the 24-48 GB VRAM limit of existing machines, showing that QLoRA is indeed a practical approach for fine-tuning large models without using multiple GPUs. The reasons why QLoRA is chosen go beyond just efficiency. Given the size of GPT-OSS-20B models, complete fine-tuning is often impossible to do in an average research lab due to heavy memory and computation requirements. QLoRA, therefore, acts both as an efficiency measure and an enabling factor making supervised adaptation possible with limited hardware.

The optimizer uses paged_adamw_8bit with bf16 precision and cosine learning rate. The micro-batch size is 1, and the gradients are accumulated to form batches of size 16 over 16 iterations. Training takes place for five epochs, and various class balancing strategies are used for HateXplain, including weighted cross entropy loss and weighted random sampling. A detailed list of the hyperparameters is given in Table 1.

Table 1. *Training Hyperparameters*

Hyperparameter	Value	Description
Learning rate	5×10^{-4}	Learning rate used in QLoRA fine-tuning
Batch size	1	Micro-batch size
Gradient accumulation	16	Effective batch size = 16
Optimizer	paged_adamw_8bit	Memory-efficient AdamW variant
Epochs	5	Number of training epochs
Warm-up ratio	0.1	Proportion of steps for LR warm-up
Precision	bf16	Numerical format for stability
Scheduler	Cosine	Learning-rate decay schedule

IV. EXPERIMENTS

A. DATASET

HateXplain [6] is a multiannotator dataset designed for analyzing hate speech. Each data sample contains the text of the comment, three annotations, the reasoning spans, and metadata pointing to the targets mentioned in the message. Three types of statements are distinguished: Normal, Offensive, and Hate Speech. The annotation scheme captures conflicts between annotators and draws attention to edge cases that may be difficult to classify.

Table 2. *Representative samples from the HateXplain dataset*

Attribute	Description
Total Comments	20,148
Labels	1. HATE: Content with hateful intent targeting specific groups or individuals. 2. OFFENSIVE: Content with profanity or abuse but no explicit targeting. 3. NORMAL: Content without offensive or hateful elements.
Label Distribution	HATE: 5,935 (29.45%) OFFENSIVE: 5,480 (27.22%) NORMAL: 7,814 (38.78%) UNDECIDED: 919 (4.56%)
Data Sources	Twitter: 9,055 posts Gab: 11,093 posts
Data Splits	Training: 80% (15,383 comments) Validation: 10% (1,922 comments) Testing: 10% (1,924 comments)
Annotation Process	Three annotators per comment. Final labels determined by majority voting. Undecided cases excluded.
Characteristics	Includes span annotations (rationales) highlighting text justifying the label. Focuses on diversity in hate speech types and targets. Provides robust metrics for bias and explainability evaluation.

B. EXPERIMENTAL SETUP

Experiments are conducted on the full HateXplain dataset using the official train, validation and test splits. The preprocessing stage resolves annotator disagreement, extracts target-group information and rationale cues and formats each instance into the instruction structure described in Section III.

GPT-OSS-20B is fine-tuned with QLoRA, using 4-bit NF4 quantization and low-rank adapters inserted into all linear layers. Training employs paged_adamw_8bit with bf16 precision, cosine scheduling and an effective batch size of 16 through gradient accumulation. Class imbalance is handled by weighted cross-entropy and weighted random sampling.

All experiments run on a single high-memory GPU. On hardware such as an NVIDIA RTX A6000 Ada (48 GB), VRAM usage remains around 18–22 GB during training, allowing the full 20-billion-parameter model to be fine-

tuned without multi-GPU infrastructure. The configuration produces stable convergence across the full dataset, and the resulting performance is reported in Section E.

To support reproducibility, the implementation of the proposed method is publicly available on GitHub Link: https://github.com/coderkhongodo/hateXplain_gptoss. The repository contains the main experimental components, including preprocessing, instruction construction, fine-tuning configuration, and evaluation scripts.

C. COMPARED METHODS

The purpose of the comparison in this study is to position the proposed method against representative modeling paradigms for hate speech detection rather than to establish a strictly equivalent experimental setting across all systems. Specifically, the comparison is designed to reflect four practical directions commonly used in the literature, namely traditional lightweight neural models, supervised transformer encoders, parameter-efficient fine-tuning of large language models, and prompting-based LLM inference. Because these paradigms differ substantially in architecture, training strategy, and adaptation mechanism, the comparison should be interpreted as a cross paradigm performance assessment on the same benchmark rather than a fully controlled like-for-like evaluation.

To evaluate the effectiveness of the proposed GPT-OSS-20B model, we conducted a comparative study against a representative set of multilingual hate-speech classification baselines. Two fastText-based neural architectures (TextCNN and GRU) were included as traditional lightweight baselines. A group of multilingual transformer encoders bert-base-multilingual-uncased, bert-base-multilingual-cased, distilbert-base-multilingual-cased, and xlm-roberta-base were fine-tuned directly on the HateXplain dataset under a unified training protocol, using a maximum sequence length of 50, the AdamW optimizer with a learning rate of 2×10^{-5}, 4 epochs, and a batch size of 16, with XLM-R additionally employing a dropout rate of 0.1. The detailed hyperparameter settings of these supervised transformer baselines are summarized in Table 3. We also include the BERT-HateXplain model augmented with LIME [6] rationales as an interpretable supervised baseline.

The comparison further incorporates inference-only large-language-model baselines, including Flan-T5-Large [10], GPT-3.5-turbo-0301 [10], and GPT-4o-mini evaluated under zero-shot and few-shot prompting configurations. All systems were assessed on the same HateXplain test split and compared using accuracy, precision, recall, and macro-F1 to ensure full consistency across the experimental evaluation.

Table 3. *Hyperparameter configuration of the baseline models*

Hyperparameter	Value	Description
Maximum sequence length	50	Maximum input sequence length used for fine-tuning
Optimizer	AdamW	Optimization algorithm used for training
Learning rate	2×10^{-5}	Learning rate applied to baseline training
Epochs	4	Number of training epochs
Batch size	16	Batch size used for training
Dropout (XLM-R only)	0.1	Additional dropout applied only to XLM-RoBERTa

D. EVALUATION METRICS

Below are the measures used to evaluate the model's performance:

Accuracy, Macro F1-Score, Precision and Recall. For those datasets that have a very pronounced class imbalance problem, the Macro F1-Score is given significant importance. The reason why Macro F1-Score should be used is because unlike Accuracy measure, which may favor the larger class due to its biasness, Macro F1-Score gives equal importance to both classes (toxic/non-toxic).

E. RESULTS AND DISCUSSION

Table 4. *Performance comparison between the proposed model and baseline methods*

Model	Accuracy (%)	Precision (%)	Recall (%)	mF1 (%)
Text CNN fastText	0.6377	0.6187	0.6041	0.6341
GRU fastText	0.6133	0.5979	0.5833	0.6133
mBERT (uncased)	0.6517	0.6448	0.6301	0.6601
mBERT (cased)	0.6569	0.6503	0.6356	0.6656
mDistilBERT (cased)	0.6606	0.6542	0.6395	0.6695
XLM-RoBERTa (base)	0.6622	0.6544	0.6397	0.6697
BERT-HateXplain [6]	0.6980	-	-	0.6870

Flan-T5-Large [10]	0.6000	0.7000	0.6500	0.5900
GPT-4o-mini Zero-shot	0.6297	0.5912	0.5673	0.6173
GPT-4o-mini Few-shot	0.6894	0.6678	0.6437	0.6937
GPT-OSS-20B (Ours)	**0.8337**	**0.8248**	**0.8099**	**0.8066**

Comparison between the proposed model and baselines in terms of their performances is illustrated in Table 4. It shows that the proposed model yields a macro F1 of 80.66% and accuracy of 83.37%, significantly surpassing all baselines based on fastText classifiers and multilingual transformers, especially in their supervised counterparts BERT-multilingual and XLM-RoBERTa. Such an improvement indicates that the model can generalize well for the diverse annotation scheme of HateXplain while providing stable results across the whole spectrum of categories.

The confusion matrix shown in Figure 1 also supports this conclusion. The model manages to correctly classify most of the samples in each class with high recall in cases of non-toxic and implicit-hate classes, which proves that the model does not exhibit overpredictions of the majority class, a common problem for some baselines. In case of wrong predictions, the misclassified samples include samples with implicit hate expressed through sarcastic phrases, idioms, and other complex language patterns, which are difficult for both fine-tuned and large-size transformers to capture.

Comparing the performance of the proposed GPT-OSS-20B model with LLMs used as baselines, the latter significantly lag behind. Specifically, the difference between GPT-3.5-turbo-0301 and the proposed solution is very clear. Additionally, the proposed approach outperforms even fine-tuned version of GPT-4o-mini, which provides only 69.37% macro F1, being used in the few-shot fashion. In contrast to these baselines, the proposed approach benefits from stable optimization and is not constrained with language-specific preprocessing, typical of transformers. Thus, the performance of GPT-OSS-20B on the HateXplain dataset is rather strong yet still leaves room for improvement due to errors in cases of nuanced language patterns, like cultural slang, emojis, and indirect aggression.

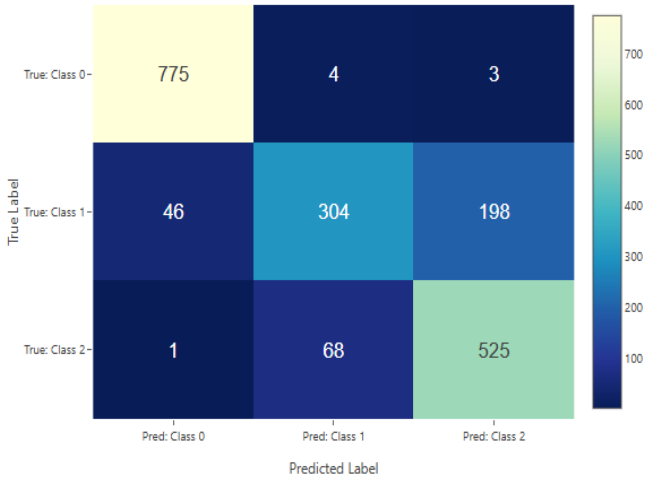

Figure 2. *Confusion matrix of the proposed method on the HateXplain dataset*

V. CONCLUSION

This paper proposes an instruction-tuned framework for the detection of hate speech using GPT-OSS-20B trained with QLoRA to allow efficient tuning of the model. The framework utilizes the concept of annotator-aware label consolidation, rationale-preserving pre-processing, and class-balanced training to enable detection of contextual and implied harm in the context of the HateXplain data. Experimental findings suggest that the proposed solution attains competitive performance compared to existing transformer baseline models and nears the effectiveness of current large language models that rely on prompts, all while using just a single GPU with ample memory capacity. Limitations associated with the study include the reliance on one dataset and lack of cross-domain evaluations. These are mitigated to some extent via the use of balanced sampling techniques and resource analysis. Possible future research directions include the development of a framework capable of handling multiple datasets, the inclusion of rationale-based supervision, and the exploration of more efficient parameter tuning methods.

VI. REFERENCES

[1] M. Subramanian, V. Easwaramoorthy Sathiskumar, G. Deepalakshmi, J. Cho, and G. Manikandan (Oct. 2023), "A survey on hate speech detection and sentiment analysis using ma-chine learning and deep learning models," Alex. Eng. J., vol. 80, pp. 110–121, doi: 10.1016/j.aej.2023.08.038.

[2] H. M. Raza Ur Rehman, M. Saleem, M. Z. Jhandir, E. S. Alvarado, H. Garay, and I. Ashraf (May. 2025), "Detecting hate in diversity: a survey of multilingual code-mixed image and video analysis," J. Big Data, vol. 12, no. 1, p. 109, doi: 10.1186/s40537-025-01167-w.

[3] B. Barakat and S. Jaf (Aug. 2025), "Beyond Traditional Classifiers: Evaluating Large Language Models for Robust Hate Speech Detection," Computation, vol. 13, no. 8, p. 196, doi: 10.3390/computation13080196.

[4] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer (2023), "QLORA: efficient fine-tuning of quantized LLMs," in Proceedings of the 37th International Conference on Neural Information Processing Systems, in NIPS '23. Red Hook, NY, USA: Curran As-sociates Inc.

[5] M. U. Arshad and W. Shahzad (Oct. 2024), "Understanding hate speech: the HateInsights dataset and model interpretability," PeerJ Comput. Sci., vol. 10, p. e2372, doi: 10.7717/peerj-cs.2372.

[6] B. Mathew, P. Saha, S. M. Yimam, C. Biemann, P. Goyal, and A. Mukherjee (May. 2021), "HateXplain: A Benchmark Dataset for Explainable Hate Speech Detection," Proc. AAAI Conf. Artif. Intell., vol. 35, no. 17, Art. no. 17, doi: 10.1609/aaai.v35i17.17745.

[7] E. W. Pamungkas and V. Patti (2019), "Cross-domain and Cross-lingual Abusive Language Detection: A Hybrid Approach with Deep Learning and a Multilingual Lexicon," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguis-tics: Student Research Workshop, Florence, Italy: Association for Computational Lin-guistics, pp. 363–370. doi: 10.18653/v1/P19-2051.

[8] M. Usman, M. Ahmad, G. Sidorov, I. Gelbukh, and R. Q. Tellez (July. 2025), "A Large Language Model-Based Approach for Multilingual Hate Speech Detection on Social Media," Computers, vol. 14, no. 7, p. 279, doi: 10.3390/computers14070279.

[9] Y. Xiao, Y. Hu, K. T. W. Choo, and R. K.-W. Lee (Oct. 2024), "ToxiCloakCN: Evaluating Ro-bustness of Offensive Language Detection in Chinese with Cloaking Perturbations," in Proceedings of the 2024 Conference on Empirical Methods in Natural Language Pro-cessing, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Asso-ciation for Computational Linguistics, pp. 6012–6025. doi: 10.18653/v1/2024.emnlp-main.345.

[10] S. Roy, A. Harshvardhan, A. Mukherjee, and P. Saha (2023), "Probing LLMs for hate speech detection: strengths and vulnerabilities," in Findings of the Association for Computa-tional Linguistics: EMNLP 2023, Singapore: Association for Computational Linguistics, pp. 6116–6128. doi: 10.18653/v1/2023.findings-emnlp.407.

[11] L. Wang et al. (May. 2025), "Parameter-efficient fine-tuning in large language models: a survey of methodologies," Artif. Intell. Rev., vol. 58, no. 8, p. 227, doi: 10.1007/s10462-025-11236-4.

ĐÁNH GIÁ MÔ HÌNH GPT-OSS-20B CHO PHÁT HIỆN NỘI DUNG THÙ GHÉT: CẢI TIẾN THÍCH NGHI HIỆU QUẢ THAM SỐ

Lê Hồng Quang, Huỳnh Lý Tân Khoa, Thanh Lê

TÓM TẮT— Việc phát hiện ngôn từ thù hận tiếp tục đặt ra các thách thức về phương pháp do sự mơ hồ trong gán nhãn, sự mất cân bằng lớp và yêu cầu phân biệt chi tiết giữa các biểu đạt mang tính xúc phạm và mang tính thù hận. Nghiên cứu này thực hiện một phương pháp điều chỉnh tiết kiệm tham số cho một mô hình ngôn ngữ lớn quy mô 20 tỷ tham số nhằm thực hiện phân loại ngôn từ thù hận ba lớp. Cách tiếp cận đề xuất hợp nhất các quyết định của người gán nhãn thành một nhãn duy nhất cho mỗi mẫu, áp dụng lấy mẫu cân bằng để giảm mức độ thưa thớt của lớp thiểu số, và tích hợp các mẫu chỉ dẫn cùng siêu dữ liệu dựa trên mức độ đồng thuận nhằm ổn định dự đoán trong các trường hợp mơ hồ giữa các lớp. Mô hình sau điều chỉnh được đánh giá đối sánh với các mô hình nền dựa trên transformer encoder và các cấu hình mô hình ngôn ngữ lớn theo hướng prompting. Kết quả cho thấy hệ thống được tinh chỉnh đạt macro F1-score 80.66% và độ chính xác 83.37%, vượt trội hơn so với tất cả các mô hình nền so sánh, với mức cải thiện đặc biệt mạnh ở hạng mục Hate Speech. Phân tích bổ sung về mức sử dụng tính toán cho thấy mô hình được tinh chỉnh vận hành trong điều kiện tài nguyên ở mức vừa phải. Các kết quả này cho thấy điều chỉnh tiết kiệm tham số theo hướng gọn nhẹ là một lựa chọn khả thi cho phân loại hate speech chi tiết khi việc tinh chỉnh toàn phần các mô hình ngôn ngữ lớn không khả thi.

Từ khóa— Phát hiện ngôn từ thù hận; Mô hình ngôn ngữ lớn; Tinh chỉnh QLoRA; Tinh chỉnh theo chỉ dẫn.

Lê Hồng Quang tốt nghiệp chuyên ngành Hệ thống thông tin thuộc ngành Công nghệ thông tin tại Trường Đại học Ngoại ngữ - Tin học TP. Hồ Chí Minh (HUFLIT), Việt Nam vào năm 2021. Anh nhận bằng thạc sĩ Công nghệ Thông tin tại HUFLIT vào năm 2026. Hiện anh đang công tác tại Phòng Chính trị - Công tác sinh viên HUFLIT. Lĩnh vực nghiên cứu, quan tâm bao gồm: ứng dụng công nghệ trí tuệ nhân tạo (AI) trong quản lý giáo dục và Xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP).

Lê Thanh tốt nghiệp ngành Công nghệ Thông tin tại Trường Đại học Công nghệ TP. Hồ Chí Minh (HUTECH), Việt Nam vào năm 2011. Anh nhận bằng thạc sĩ Công nghệ Thông tin tại HUTECH vào năm 2018. Sau đó, anh tiếp tục theo học chương trình tiến sĩ chuyên ngành Xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP) tại HUTECH. Hướng nghiên cứu chính của anh bao gồm: Học sâu, Khoa học dữ liệu và Xử lý ngôn ngữ tự nhiên. Hiện nay, anh đang công tác với vai trò giảng viên tại Trường Đại học Kinh tế – Tài chính TP. Hồ Chí Minh (UEF).

Huỳnh Lý Tân Khoa tốt nghiệp chuyên ngành Khoa học Dữ liệu thuộc ngành Công nghệ Thông tin tại Trường Đại học Ngoại ngữ - Tin học Thành phố Hồ Chí Minh (HUFLIT), Việt Nam vào năm 2026. Lĩnh vực nghiên cứu, quan tâm bao gồm: Học sâu, Khoa học dữ liệu và Xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP).