Use of Multimodal Large Language Models to Detect Hate Speech in Social VR

Pingqi An
Khoury College of Computer Science
Northeastern University
Vancourver, Canada
an.p@northeastern.edu

Teng Liu
Khoury College of Computer Science
Northeastern University
Vancourver, Canada
liu.ten@northeastern.edu

Yuhao Lu
Khoury College of Computer Science
Northeastern University
Vancourver, Canada
lu.yuhao@northeastern.edu

***Abstract*—As virtual reality continues to advance, the issue of hate speech on these platforms is becoming increasingly serious. This study focuses on enhancing hate speech detection in social VR platforms using our SLAM-HATE-SPEECH model, a Multimodal Large Language Model that integrates audio and textual data. We adapted the HateMM dataset to include over 500 audio files, with 50% labeled as hate speech and clipped to 30 seconds to standardize inputs. Our research evaluates SLAM-HATE-SPEECH model's effectiveness in various configurations: audio-only, text-only, and combined audio-text analysis. This approach aims to improve the accuracy and adaptability of hate speech detection in VR environments.**

I. Introduction

VR platforms have gained popularity due to their immersive experiences that offer users interactive and lifelike environments, revolutionizing how people connect and engage in social activities online. However, hate speech is also becoming more common on these platforms. Hate speech is public speech that expresses hate or encourages violence towards a person or group based on something such as race, religion, sex, or sexual orientation [1]. Hate speech not only affects the user experience but can also harm users' mental health. In 2019, Hate Crime Statistics showed 15,588 law enforcement agencies reported crimes, suspects, offenders, and hate crime zones[2]. These groups reported 7,314 hate crimes with 8,559 offenses. By improving the detection capabilities for hate speech, platforms can more effectively identify and address such misconduct, thereby enhancing the safety and satisfaction of all users. Therefore, it is important to develop a system that can effectively detect hate speech in these immersive environments.

Previous research has primarily concentrated on employing large language models to analyze textual contents of speech for hate speech detection. These models have been effective for detecting hateful language in social media platforms like Twitter, but they have limitations when applied to environments where voice communication is the primary mode of interaction [3]. Specifically, these methods often neglect the audio features of speech, which play a crucial role in detecting the tone and emotion of the speaker [4].

Given the limitations identified in previous research on large language models (LLMs), especially in social VR environments dominated by voice interactions, it becomes imperative to explore more comprehensive approaches. Multimodal Large Language Models (MLLMs) can integrate audio and textual data, enhancing language understanding. And because we now have two modalities, we can also perform cross-validation, which increases the accuracy of hate speech detection. MLLMs like SLAM-ASR [5], effective in converting spoken language to text, have yet to be widely applied in VR.This presents an opportunity to employ these technologies in VR environments, where solid solutions are needed for detecting hate speech.

For our research, we will investigate the use of MLLMs, particularly SLAM-ASR [5], to leverage the combined strengths of textual and audio content. This approach aims to enhance both the accuracy and efficiency of detecting hate speech. We are focusing on both 'what is said' (text content) and 'how it is said' (audio features). This integration allows for a more comprehensive detection of hate speech, capturing details that text-only or audio-only methods might overlook, especially in dynamic VR environments.

II. Literature Review

A. Hate Speech

Hate speech identification is a highly contextual problem, and current detection methods are severely limited in their ability to accurately capture the this context [6]. Hate speech has grown to be a serious issue on all social media platforms, contributing to a number of hate crimes and negatively impacting the mental and emotional health of those who are targeted [7]. This emphasizes the need for techniques to identify hate speech on the internet [8]. Mukherjee et al. [9] attempted to solve this problem using both classical and deep learning approaches. They found that both methods have drawbacks in terms of the need for extensive handcrafted features, model architecture design, and pretrained embeddings that have limitations at capturing semantic relationships between words.

B. LLMs for Hate Speech Detection

Guo et al. [6] provide a comprehensive review of several LLMs like ChatGPT [10], including their performance on various benchmarks. Their research highlights the importance of model design, particularly the use of prompts, in improving the accuracy of hate speech detection. There are four introduced

prompts, general prompt, general prompt with hate speech definition, few-shot learning prompt and CoT prompt. For instance, few-shot learning prompts involve training a model with a minimal number of examples to optimize its responses, thereby allowing the LLMs to adapt quickly with limited data. By experimenting with various prompting strategies, such as few-shot learning prompts to optimize LLMs responses, the paper sheds light on how subtle changes in prompting methods can significantly affect hate speech detection outcomes. This variability highlights the need for customized prompting strategies, such as CoT prompting, that are better suited to the complex demands of hate speech detection.

C. MLLMs

Li et al. [11] describe MLLMs as a significant advancement in AI technology, capable of processing and generating content across multiple modalities, such as text, images, audio, and video. These models surpass traditional unimodal systems by integrating diverse data inputs, allowing for a more comprehensive understanding of complex information. This holistic approach is particularly beneficial in tasks that require reasoning and decision-making, such as medical diagnostics, autonomous systems, and robotics. Li et al. also emphasize the versatility of MLLMs, noting their ability to adapt to various specialized domains and industries. Their research is highly beneficial for our hate speech detection project, as it demonstrates the advantages of MLLMs compared to traditional unimodal systems.

D. Improved ASR for MLLMs

The integration of MLLMs into virtual reality environments for hate speech detection represents a significant advance in handling complex multimedia content. A crucial component in enhancing MLLMs performance is the optimization of Automatic Speech Recognition (ASR) systems to accurately transcribe and analyze spoken language.

In 2024, Ma et al. [5] challenged the prevailing complexity in LLM-based ASR systems. They advocated for a streamlined approach that involved a single trainable linear projector along with a fixed LLM and speech encoder, contrary to more complex designs that might include multiple trainable layers and sophisticated alignment mechanisms. Their SLAM-ASR model, leveraging components like HuBERT X-Large [12] and Vicuna-7B [13], did achieve superior performance on standard benchmarks like LibriSpeech. By using a fixed speech encoder and LLM, their system benefits from stability and reduces the potential for overfitting, which is common in more elaborate setups. The SLAM-ASR model presented demonstrates considerable improvements in word error rates (WER) compared to traditional ASR models, highlighting the efficacy of integrating well-tuned, pre-existing models over building complex new systems from scratch. Building on their research, we will continue to refine and optimize the design to further enhance its applicability to hate speech detection in virtual reality environments.

E. Limitations and Gaps

From our observations, the field of hate speech detection has predominantly focused on analyzing data through a single modality, either audio or text. Despite some advancements in leveraging MLLMs and their application within social VR environments, significant challenges persist. These modalities, when utilized separately, often fail to capture the full spectrum of communicative cues necessary for accurate hate speech detection. As research progresses, there is a compelling need to enhance the integration of MLLMs and VR to better understand and interpret the complex interactions in VR settings. Moreover, efforts such as those by Guo et al. [6] in standardizing prompting strategies for LLMs highlight the ongoing need to refine these approaches to improve detection accuracy and system adaptability in real-world applications.

III. Problem Statement

The main problem addressed in this research is the effective detection of hate speech in social VR. The challenge of this project lies in the need to integrate textual and auditory data through MLLMs. Previous research has focused on unimodal approaches (text or audio), resulting in significant gaps in effectively handling the complexity of real-time interactions.

We are utilizing the modified HateMM [14] dataset, which includes over 500 audio files, with approximately 50% labeled as hate speech and 50% as non-hate speech. Each file is adjusted to a maximum duration of 30 seconds using audio clipping techniques. This modification makes it highly suitable for training models to accurately identify and respond to hate speech.

In our implementation, we opted to use a multimodal large language model built on SLAM-LLM (Speech, Language, Audio, and Music processing with Large Language Model) [15]. Building upon this, we applied code modifications and retraining using SLAM-ASR— a model capable of generating transcriptions from user-input audio datasets. The aim is to achieve hate speech detection by simultaneously integrating both the target audio and its corresponding transcription. In this context, the retrained model will be capable of simultaneously analyzing both the acoustic features in the audio and sensitive keywords in the text. Therefore, our hypothesis is that this multimodal large language model will demonstrate superior accuracy compared to a unimodal model. Ultimately, we aim to integrate our MLLMs into a Social VR environment to demonstrate its effectiveness in detecting hate speech. Therefore, we named our trained model SLAM-HATE-SPEECH. We are building and evaluating the model from three directions, for which we have designed three experiments:

(1) Provide only the audio, without transcription, to the model and assess its detection efficiency, as shown in Figure 1.

(2) Running a pre-existing transcription model Whisper [16] to generate the audio transcription, using the transcription as a prompt for the model, and measuring its detection efficiency, as shown in Figure 2.

Fig. 1. SLAM-HATE-SPEECH detection process. This process receives only audio input.

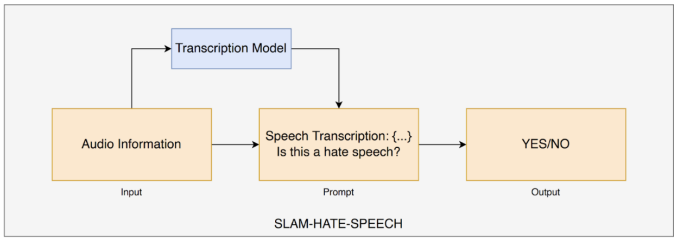

Fig. 2. SLAM-HATE-SPEECH detection process using Whisper for transcription.

(3) Running the transcription model Whisper to generate the audio transcription, including the transcription in the prompt for the model without any additional content, and measuring its detection efficiency, as shown in Figure 3.

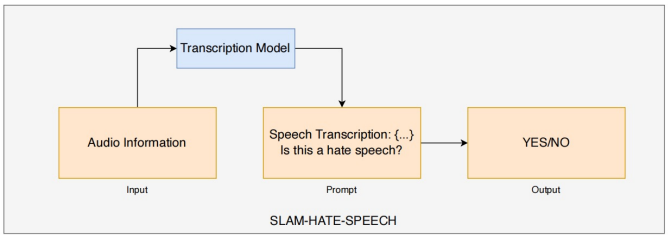

Fig. 3. SLAM-HATE-SPEECH test-based detection process.

Each experiment aims to assess the effectiveness of the multimodal approach using precision, accuracy, recall, and F1-score[17] as our key metrics, providing a comprehensive evaluation of the model's performance from multiple perspectives.

The primary metric we will use is the F1-score. As shown in the equation below, the F1-score is the harmonic mean of precision and recall. Precision (the proportion of correctly identified hate speech cases out of all predicted hate speech cases) and recall (the proportion of correctly identified hate speech cases out of all actual hate speech cases) are both critical in evaluating the model's ability to accurately detect hate speech.

$$F_1 = \frac{2}{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}} = 2\frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}.$$

The F1-score balances the trade-off between precision and recall, making it especially useful when dealing with imbalanced datasets, where false positives and false negatives may have different consequences. By focusing on the F1-score, we can assess the model's ability to correctly identify hate speech while minimizing both false positives and false negatives, which is crucial for our application. A higher F1-score indicates a better overall performance in detecting hate speech with balanced accuracy.

IV. Methodology

A. Model Architecture

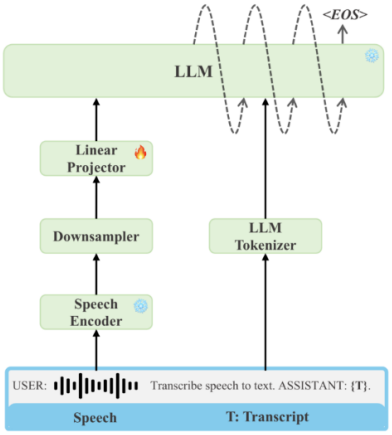

Fig. 4. Architecture of SLAM-ASR [5]

To introduce our model architecture, it is necessary to first provide a detailed overview of our base model, SLAM-ASR, along with its functionalities and architectural structure. As shown in Figure 4, this is the model architecture of SLAM-ASR. It was originally designed for Automatic Speech Recognition, and its key architectural feature is the use of a frozen speech encoder and a frozen LLM. This means that within the SLAM-ASR framework, these two components remain untrainable. SLAM-ASR enables users to focus on training a trainable linear projector, which aligns the speech and text modalities.

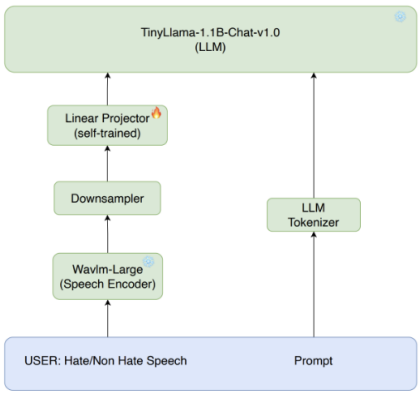

Fig. 5. Architecture of SLAM-HATE-SPEECH

Our model SLAM-HATE-SPEECH is built on the SLAM-ASR architecture, where we train a custom linear projector with specialized functionalities. We will train this linear projector using labeled hate speech and non-hate speech data

across three experiments. By comparing and analyzing the outcomes of these three training approaches, we aim to develop a MLLM that is optimally suited for hate speech detection in Social VR environments.

In the original SLAM-ASR framework, it used WavLM-Large [18] as the encoder and Vicuna-7B as the LLM. We transitioned from the Vicuna-7B model to TinyLlama [19] for the fine-tuning process of our SLAM-HATE-SPEECH model. TinyLlama offers several advantages, particularly in terms of efficiency and adaptability. As a smaller variant of the larger LLMs, TinyLlama requires significantly less computational resources, which facilitates faster training and inference times. This efficiency makes it ideal for real-time applications, such as moderating discussions in social VR environments where quick response times are crucial. Furthermore, TinyLlama retains much of the robust language understanding capabilities of its larger counterparts, making it highly effective even with its reduced size. This efficiency does not come at the cost of performance, as TinyLlama continues to deliver high accuracy in language tasks. Additionally, our model still incorporates the WavLM-large encoder, chosen for its superior ability to process complex audio signals.

B. Experimental Setup

(1) Dataset

Modified HateMM Dataset: The HateMM dataset is a specialized resource for training LLMs to identify hate speech. This original dataset includes over 200 audio files, with 50% labeled as containing hate speech and 50% as non-hate speech. Due to the original length of many files exceeding 30 seconds, audio clipping techniques were applied to adjust each segment to a maximum duration of 30 seconds. The modified dataset contains over 500 audio files. This modification ensures uniformity and consistency across the dataset, making it highly suitable for robust hate speech detection training.

(2) Training Procedure

The training process of SLAM-HATE-SPEECH consists of three stages: audio-only detection, integrated multimodal detection, and text-only detection. Each experiment builds upon the previous one, progressively incorporating more information to evaluate the impact of different input modalities on hate speech detection.

2.1 Experiment 1: Audio-Only Input

In this experiment, we aim to evaluate SLAM-HATE-SPEECH's performance when provided with audio data, without any transcription. This stage leverages the model's ability to extract acoustic features and detect hate speech purely from the auditory modality. The model processes the raw audio input, and based on its acoustic patterns, determines whether the content qualifies as hate speech.

The linear projector is trained to map the audio features to hate speech or non-hate speech labels, optimizing for classification accuracy. And also this serves as a baseline to assess the limitations of audio-only hate speech detection.

2.2 Experiment 2: Transcription from Whisper

In the second experiment, we will formally incorporate multimodal detection into our model. Unlike the first experiment, which only uses audio as input, we will employ an external ASR model, Whisper, to generate transcriptions corresponding to the audio input. In this case, we will combine these transcriptions as prompts for SLAM-HATE-SPEECH, along with the audio data as input. As a multimodal detection approach, the objective of this experiment is to determine how incorporating transcriptions generated by a third-party ASR model impacts the model's detection performance.

The linear projector will be trained to combine the audio and text inputs, aligning them to improve hate speech detection. The model will optimize for multimodal input fusion, learning to utilize both acoustic and textual features effectively.

2.3 Experiment 3: Text-Only input

In this third experiment, we aim to evaluate the performance of the SLAM-HATE-SPEECH model when provided solely with text data, specifically using transcriptions generated by the Whisper model from the audio inputs. This approach allows us to focus exclusively on the model's ability to detect hate speech through text analysis.

The linear projector in this setup will be trained to map these textual features to hate speech or non-hate speech labels, focusing on optimizing text-based classification accuracy. This experiment serves as a crucial test of the model's capabilities in scenarios where audio is not available or is impractical to use. It also helps to establish a baseline for comparing the effectiveness of text-only versus multimodal approaches in complex real-world environments like social VR.

C. Data Augmentation

Since the hate speech dataset available to us is limited, we employ data augmentation techniques to ensure the robustness and generalizability of our model. By artificially expanding the dataset through various transformations, we aim to enhance the model's ability to detect hate speech in diverse acoustic conditions.

In our approach, we apply several audio augmentation techniques to introduce variability into the training data. These transformations include adding noise, time stretching, and pitch shifting, which are commonly used to simulate real-world variations in speech. Specifically:

(1) Noise Addition

In this process, random Gaussian noise is added to the audio signal, which introduces small, random variations in the sound. This technique simulates real-world situations where background noise or interference might occur during speech recording (e.g., environmental noise, microphone imperfections). By training the model with noisy data, it becomes more robust to such disturbances, improving its ability to detect hate speech in less-than-ideal recording conditions and ensuring the model is not overly sensitive to clean, noise-free input.

(2) Time Stretching

Time stretching involves modifying the speed of the audio without affecting its pitch. This transformation stretches or compresses the duration of the audio, mimicking variations in

speech rate, such as a speaker talking faster or slower. Time stretching is particularly useful for exposing the model to a wider range of speech patterns, helping it adapt to different speaking tempos and rhythms. This can improve the model's performance when faced with speech that varies in pace, such as fast-talking or slow and deliberate speech.

(3) Pitch Shifting

Pitch shifting alters the frequency of the audio, either raising or lowering the pitch of the speech. This simulates changes in a speaker's vocal characteristics, such as variations in tone, accent, or emotional state. For example, pitch shifting can help the model become less sensitive to individual speaker characteristics, enabling it to focus more on the content of the speech rather than the speaker's voice. This is particularly important for hate speech detection, as the tone and pitch of speech can vary significantly across different individuals, yet the model needs to detect harmful content consistently regardless of these variations.

These techniques are applied with a probability of 50% for each transformation, allowing for a diverse range of augmented samples.

D. Evaluation Metrics

The primary metric for evaluating the performance of our hate speech detection model is the F1-score. This measure is crucial as it balances precision and recall, reflecting the model's ability to correctly identify hate speech (true positives) while minimizing false positives (non-hate speech misclassified as hate speech) and false negatives (hate speech missed by the model). The emphasis on the F1-score is particularly significant in the model's operational context, where both false positives and false negatives are costly, and a balanced performance is essential for effective moderation in dynamic communication environments, especially in social VR. A higher F1-score indicates that the model is performing well in accurately and consistently detecting hate speech while minimizing errors.

V. Results

A. Model Fine-tuning Analysis

We designed 50 epochs in total, consisting of 2.5k steps, with each training batch involving 8 audio files, for the fine-tuning process across all three experiments. This setup is sufficient for us to observe the changes in key metrics during training, including training accuracy, training loss, validation accuracy, and validation loss. The best fine-tuned model for our hate speech detection task is selected based on the highest validation accuracy. This criterion is chosen because validation accuracy serves as a reliable indicator of the model's ability to generalize to unseen data, ensuring it is not overfitting to the training set and performs effectively on real-world inputs.

(1) Fine-Tuning Process in Experiment 1

For Experiment 1, we focused on evaluating the model's performance when only speech was provided as input. The prompt used in this experiment was:

Analyze the tone, pitch, and speech patterns in the audio. Pay attention to aggressive, hostile, or discriminatory tones, along with any hateful or harmful word choices in the transcription (e.g., racial slurs or insults). If hate speech is detected in either the audio or transcription, respond 'Yes'; otherwise, respond 'No'. Ensure the response is strictly 'Yes' or 'No' without additional content.

In this setup, the LLM receives input consisting of audio features (processed by an encoder) combined with the specified prompt, effectively creating a multimodal approach where both audio and textual elements (the prompt content) are involved. The key significance of this experiment lies in directing the model to analyze hate speech solely based on audio signals, without additional contextual information. Consequently, the model's performance is evaluated strictly on its ability to detect hate speech by analyzing audio attributes such as tone, pitch, and speech patterns, guided by the provided prompt.

Below, Figure 6 shows the key metrics during the fine-tuning process of Experiment 1. Despite using audio-only input in this approach, both the training loss and validation loss decrease as expected throughout the fine-tuning process. This trend indicates that our training architecture provides a solid baseline for model optimization and that the model successfully learns meaningful patterns from the data. By the end of the process, the validation accuracy stabilizes around 0.95, and the training accuracy approaches 1.0, demonstrating effective learning. Additionally, the final training and validation losses are low, further supporting the effectiveness of the fine-tuning setup.

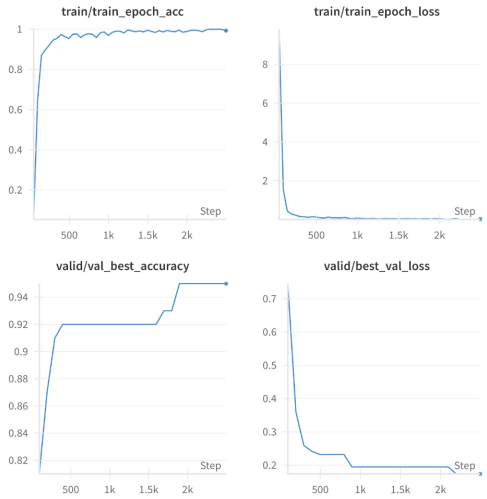

Fig. 6. Key Metrics During Experiment 1 Fine-Tuning

(2) Fine-Tuning Process in Experiment 2

In Experiment 2, we implement a truly multimodal approach. Building on the audio input from Experiment 1, we additionally utilize a transcription model to generate text transcriptions of the audio content. These transcriptions are then incorporated into the prompt provided to the model. This

setup qualifies as a genuine multimodal approach because the model processes both audio features and textual content. By combining these two modalities, the model can analyze hate speech more comprehensively—leveraging audio features such as tone and pitch while also identifying specific word choices in the text. This integration enhances the model's ability to detect hate speech effectively, highlighting the importance of both modalities in this approach. The prompt used in this phase is as follows:

The transcription of the audio is: {transcribed_text}

Evaluate both the audio and transcription. Consider tone, speech patterns, and word choices such as hateful, harmful, or aggressive language, including explicit slurs or discriminatory terms. If hate speech is detected in either, respond 'Yes'; otherwise, respond 'No'. Ensure the response is strictly 'Yes' or 'No' with no additional content.

In this phase, by providing both audio and its transcription to the model and instructing it to analyze hate speech using this multimodal input, we gain a new perspective on evaluating detection performance and assess whether this approach leads to an improved detection rate. Figure 7 below illustrates the key metrics during the fine-tuning process of Experiment 2.

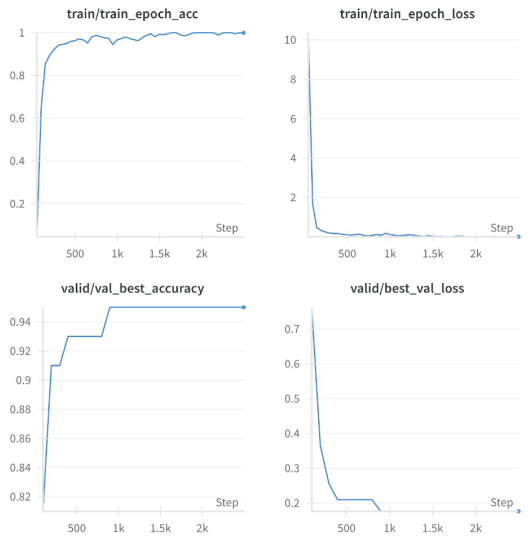

Fig. 7. Key Metrics During Experiment 2 Fine-Tuning

As shown, both training accuracy and validation accuracy demonstrate a steady upward trend over the training steps, similar to Experiment 1. However, compared to Experiment 1, the combination of audio and its corresponding transcription allows the model to learn the logic for hate speech detection more rapidly. This is evident from the sharp increase in validation accuracy during the early training stages. Furthermore, with multimodal input, the training loss and validation loss reach significantly lower levels compared to Experiment 1, further indicating enhanced detection performance achieved through this approach.

(3) Fine-Tuning Process in Experiment 3

In our final experimental setup, Experiment 3, we shifted to a text-only modality. Instead of providing audio content as input, as in the first two experiments, we disabled the audio input and relied solely on a transcription model to generate text from the audio. This transcription was included in the prompt, which was then provided to the model to detect hate speech. The prompt used in this experiment is as follows:

The transcription of the audio is: {transcribed_text}

Analyze word choices like 'hateful,' 'harmful,' 'aggressive,' or explicit slurs in the transcription. Pay attention to discriminatory terms (e.g., racial slurs or hate-related language) and assess the overall tone for hostility or intent to harm. If hate speech is detected in the transcription, respond 'Yes'; otherwise, respond 'No'. Provide only 'Yes' or 'No' without extra content.

The following Figure 8 illustrates the key metrics during the fine-tuning process of Experiment 3. Notably, in this experiment, we observe a rapid decline in validation loss, indicating that the model, using a simpler text-based detection method, is able to quickly adapt to the limited patterns available in the text data. However, due to the absence of audio features as a reference, the validation accuracy in Experiment 3 shows a much slower increase compared to the previous experiments. This slower progression reflects the model's limited ability to detect hate speech without the additional audio information.

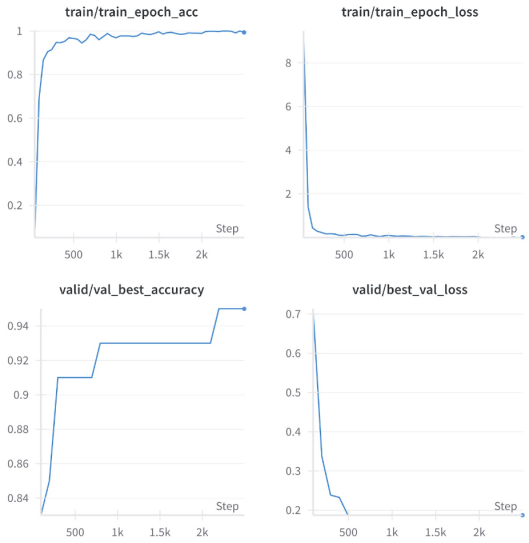

Fig. 8. Key Metrics During Experiment 3 Fine-Tuning

B. Model Performance Analysis

The experimental results obtained through 2-fold cross-validation highlight distinct patterns in model performance across different modalities for hate speech detection. The multimodal approach (Experiment 2 - Audio and Text) achieved the best performance, with an F1-score of 0.9068 and the highest accuracy of 0.9100. These results validate that integrating audio and textual features significantly enhances detection capabilities. This approach also achieved high precision (0.9373) and strong recall (0.8800), indicating its ability to minimize false positives while maintaining detection sensitivity. The

confusion matrix corroborates these findings, showing 47 true negatives and 44 true positives, with minimal false predictions (6 false positives and 3 false negatives).

Experiment	Precision	Recall	F1 Score	Accuracy
pred_exp1	0.8875	0.8400	0.8536	0.8600
pred_exp2	0.9373	0.8800	0.9068	0.9100
pred_exp3	0.9546	0.8000	0.8700	0.8800

Fig. 9. Model Performance Evaluated Using Four Metrics

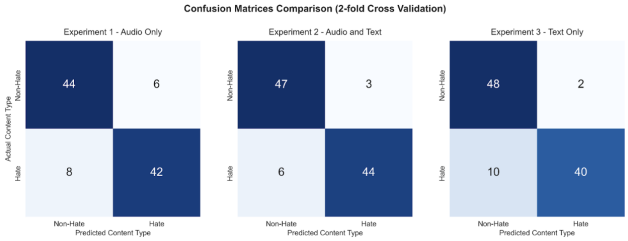

Fig. 10. Model Performance Evaluated Using Confusion Matrices

In comparison, single-modality approaches exhibited varying performance characteristics. The audio-only approach (Experiment 1) achieved an F1-score of 0.8536, with balanced but relatively lower precision (0.8875) and recall (0.8400), suggesting that audio features alone are insufficient to capture all relevant cues. The text-only approach (Experiment 3) showed high precision (0.9546) but lower recall (0.8000), resulting in an F1-score of 0.8700. This indicates that while textual analysis is effective at minimizing false positives, it may miss instances of hate speech detectable through audio cues.

The performance differences between modalities stem from the unique characteristics of each input type and their complementary nature. The multimodal approach outperforms single-modality methods by leveraging these complementary information streams, resulting in more balanced and reliable detection. This is evident in the confusion matrices, where the multimodal method demonstrates a more even error distribution compared to single-modality approaches.

VI. Discussion

The evaluation process employed for the SLAM-HATE-SPEECH model is rigorous, leveraging 2-fold cross-validation to minimize bias and ensure reliable results. The use of clear performance metrics, such as F1-score, precision, and recall, provides a robust framework for assessing model effectiveness. The inclusion of three experiments ensures that the analysis captures the strengths and weaknesses of different modalities, validating the advantages of a multimodal approach.

The results, supported by 2-fold cross-validation, provide strong evidence for the efficacy of multimodal analysis in hate speech detection. While single-modality approaches yield reasonable performance, combining multiple modalities offers a more robust and generalizable solution for real-world applications. This is particularly relevant for platforms like VRChat, where both audio and textual communication coexist, necessitating a comprehensive approach to hate speech detection across multiple channels.

VII. Conclusion

This research demonstrates that integrating audio and textual features through a MLLM significantly enhances hate speech detection in Social VR environments. The SLAM-HATE-SPEECH model achieved an F1-score of 0.9068 and an accuracy of 0.9100, outperforming single-modality approaches. These results confirm that multimodal analysis leads to more accurate and sensitive detection.

The findings have strong implications for improving user safety in VR platforms. Future research could explore extending this approach to handle longer audio clips or real-time detection scenarios. Additionally, this study was limited by the availability of hate audio data, which constrained model training and evaluation. Future efforts should also focus on developing larger and more diverse datasets, to further enhance the model's robustness and generalizability across different social VR environments.

References

[1] Cambridge Dictionary, "HATE SPEECH — meaning in the Cambridge English Dictionary," Cambridge.org, Dec. 04, 2019. https://dictionary.cambridge.org/dictionary/english/hate-speech

[2] "FBI Releases 2019 Hate Crime Statistics," Federal Bureau of Investigation, Nov. 16, 2020. https://www.fbi.gov/news/press-releases/fbi-releases-2019-hate-crime-statisticshttps://www.fbi.gov/news/press-releases/fbi-releases-2019-hate-crime-statistics

[3] Vijayaraghavan, P., Larochelle, H., Roy, D. (2019). Interpretable MultiModal Hate Speech Detection. Presented at the Machine Learning AI for Social Good Workshop, Long Beach, United States.

[4] Boishakhi, F. T., Shill, P. C., Alam, M. G. R. (2023). Multi-modal Hate Speech Detection using Machine Learning. arXiv:2307.11519.

[5] Z. Ma et al., "An Embarrassingly Simple Approach for LLM with Strong ASR Capacity." Available: https://arxiv.org/pdf/2402.08846https://arxiv.org/pdf/2402.08846

[6] Guo, K. et al. (2024) An investigation of large language models for real-world hate speech detection, arXiv.org.

[7] Chetty, N. and Alathur, S. (2018). Hate speech review in the context of online social networks. Aggression and Violent Behavior, 40(1359-1789), pp.108–118. doi:https://doi.org/10.1016/j.avb.2018.05.003.

[8] N. Prasad, S. Saha, and P. Bhattacharyya, "A Multimodal Classification of Noisy Hate Speech using Character Level Embedding and Attention," 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9533651.

[9] Mukherjee, S. and Das, S. (2024) Application of transformer-based language models to detect hate speech in social media, Journal of Computational and Cognitive Engineering.

[10] OpenAI, "ChatGPT ," ChatGPT, Nov. 30, 2022. https://chatgpt.com/

[11] M. Li et al., "Surveying the MLLM Landscape: A Meta-Review of Current Surveys," arXiv.org, 2024. https://arxiv.org/abs/2409.18991 (accessed Oct. 15, 2024).

[12] "facebook/hubert-xlarge-ls960-ft · Hugging Face," Huggingface.co, Aug. 14, 2024. https://huggingface.co/facebook/hubert-xlarge-ls960-ft (accessed Oct. 15, 2024).

[13] "lmsys/vicuna-7b-v1.5 · Hugging Face," Huggingface.co, 2023. https://huggingface.co/lmsys/vicuna-7b-v1.5 (accessed Oct. 15, 2024).

[14] "View of HateMM: A Multi-Modal Dataset for Hate Video Classification," Aaai.org, 2024. https://ojs.aaai.org/index.php/ICWSM/article/view/22209/21988 (accessed Oct. 15, 2024).

[15] X-LANCE, "GitHub - X-LANCE/SLAM-LLM: Speech, Language, Audio, Music Processing with Large Language Model," GitHub, 2023. https://github.com/X-LANCE/SLAM-LLM (accessed Oct. 15, 2024).

[16] OpenAI, “Whisper,” GitHub, Oct. 09, 2022. https://github.com/openai/whisper
[17] “F-score,” *Wikipedia*, Mar. 12, 2021. https://en.wikipedia.org/wiki/F-score
[18] “microsoft/wavlm-large · Hugging Face,” huggingface.co. https://huggingface.co/microsoft/wavlm-large
[19] “TinyLlama/TinyLlama-1.1B-Chat-v1.0 · Hugging Face,” huggingface.co, Jan. 08, 2024. https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0