MLLM Hateful Video Detection

Capstone Written Project

By

Ruijun Liu

Advisor: **Hu Hongxin**

University at Buffalo

January 2025

January 5, 2025

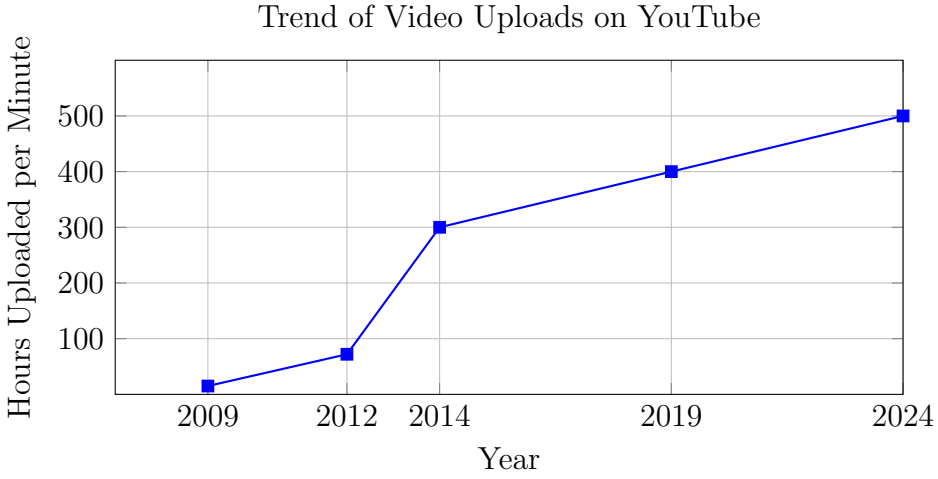
Abstract. The rise of video platforms such as Youtube, Tiktok and others has brought the surge of video content which also posted challenges to content moderations. Hateful content is one target during content moderation. Unlike traditional text-based hate speech detection, hateful video detection adds layers of complexity due to its multi-modality nature. In this study, we are exploring the strength and drawbacks of current state-of-the-art models, evaluating on how well they integrate different modalities together. We compared the methods from SOTA study on this task with multimodal large language models excel in video tasks. Our experiment results highlight that while these advanced models do benefit from multimodal inputs, they fail at utilizing audio info due to we choose Mel Spectrograms as the feature representation. A closer data reevaluation showed the most contributing factor is audio points out we need to refine current audio encoding process, and a better alignment between modalities is crucial for performances improvements.

Introduction

Since high-speed internet's widespread availability and the advancement of portable smartphones, video content has become an important part of the Internet browsing experience. According to online statistics, more than 500 hours of video are uploaded to YouTube every minute [1]. When considering other platforms such as TikTok, Vimeo, and Twitch, an estimate of more than 720000 hours of video content is uploaded on various platforms everyday [2]. However, not all of this content is in line with platform standards and mong these, a significant portion was flagged as hateful or harmful.

Hateful videos are videos that express hateful opinions towards specific ethnic or religious groups. Such content can propogate harmful stereotypes and extreme values. Furthermore, unlike hate speech which rely on text alone, hateful videos are combinations of text, audio and visual. Thus, traditional text-based approach will be inadequate for this task. Algorithmic recommendations, a prevalent feature for video platforms nowadays, will exacerbate the problem even more since they can form "echo chambers that foster radicalization among like-minded viewers" [3]. The influences of these videos can lead to real-life violence and harassment, so detecting hateful videos early-on and accurately is crucial to reducing the spread of toxic beliefs and ensuring healthy online ecosystems.

Our motivation is to conduct an exploratory research on this topic. We aim to have a better understanding of what hateful videos are, find out what can be improved upon current sota models, have a clearer picture of the main hidden obstacles and challenges preventing optimal performances, and propose and validate plausible future directions. Though hateful content detection is hard, we are fortunate enough to see the biggest rise in Multimodal Large Language Models, models integrating different data types [4]. We got promising results after adopting the pipeline provided by a study called HateMM on this topic [5], achieving 79.78% accuracy while combining three unimodal models trained on three modalities repestively, with several improvement methods. Meanwhile, we also notice excelling performances of finetuned vision LLMs such as LlaVA and VideoLlaMA. These models have been tested on video-based vision tasks and have achieved excellent results. Naturally, it came to our mind that these models might also perform well on hateul video detection. Being trained on a vast amount of vision data, models such as VideoLlaMA can describe and caption a video's content fairly well, then surely it can handle specific video types such as hateful video easily. However, we found out this is not the case. Without finetuning to hateul videos, though these vision models can still describe video content correctly, they apppear to be performing significantly worse than it's simpler counterpart (HateMM). Powerful models like Gemini-1.5-pro, LlaVA and VideoLlaMA, all having an average reduced accuracy around 10%. In the following part of this written piece, we will present and reason some interesting findings that didn't came across our mind at early stages.



Background

As we established the scale and urgency of detecting hateful videos. Now we want to define the problem quantitatively. We also highlight the key challenges that most significantly constrain performance. A clearer grasp of these underlying factors will provide the necessary foundation for our subsequent exploration.

Definition of the Problem

The biggest challenge for hateful video detection is its multimodal nature. Our goal is to utilize all three modalities in our binary classification. The problem can be described as follows: given a series of videos $\{V_1, V_2, \ldots, V_n\}$, where each video V_i is represented by three modalities $V_i = (T_i, A_i, F_i)$, corresponding to text, audio, and visual features, our detection model M is designed to map the multimodal input between two labels:

$$M : V_i \mapsto y_i \quad \text{where} \quad y_i \in \{\text{hate}, \text{non-hate}\}.$$

To solve this problem, we also need to acknowledge these issues:

Known Issues

Implicit Hate. Focusing on one modality during hateful video detection could lead to inaccurate results. For example, captions or subtitles may contain explicit hate speech yet be inadequately transcribed. Similarly, some human gestures or objects may appear entirely innocent. After a close observation of the dataset, we find frequent appearences of the US Confederate flag or a Moon-face man figure. Without proper context, these could simply be interpreted as a historical US figure and a commercial mascot. However, when combined with specific text overlays and background audio, these symbols become hateful. Our research shows that such displays are often employed by extremist groups to propagate toxic ideologies. This subtle use of symbols and memes to convey hateful messages is referred to as implicit hate.

Implicit hate adds extra layer of complexity to this task since it avoids overtly violating video platform policies. In addition to direct hate messages, sarcasm or memes have become new tools for these opinions. Within the public sphere, what was once simply difficult, perhaps impossible to notice now goes undetected all too easily. It require careful

inspection to understand the intentional message behind the video. And the nature of hateful content is always changing, creators often adapt new symbols or Internet slurs to bypass detection mechanisms. Moreover, Video platforms often host viral trends that serve as vehicles for spreading hateful ideologies, frequently disguised as humor [6, **?**]. These factors complicate the development of automated detection systems.

Lacking of sufficient datasets. Developing a comprehensive dataset to detect hateful content is difficult. Hateful content can be expressed in different languages and cultures and English datasets will be insufficient. Also, annotating such content requires a balance of inclusion and precision, because the task must be done manually, which is both time-consuming and prone to prejudice, especially when cultural and language nuances come into play [7]. The scarcity of publicly available, well-annotated datasets in several languages hampers the development of effective detection systems.

Furthermore, merely annotating hateful snippet does not allow future use of LLMs to develop essential reasoning. The dataset must also provide explicit explanations on why specific content is considered insulting. This involves connecting the textual, visual, and auditory modalities while allowing LLMs to learn through processes like chain-of-thought thinking. This type of annotation will help to improve reasoning abilities in future multimodal systems as well as present detection models.

Detection Model Limitations. Many existing approaches use unimodal analysis or basic feature fusion, thus lead to high false negative rates [**?**]. Moreover, advanced multimodal models require substantial computational resources and an enormous amount of training data.

To tackle these issues, new methodologies must incorporate powerful multimodal learning algorithms that examine text, pictures, and audio simultaneously. Robust and scalable technologies are required for detecting and controlling hateful content before it spreads, promoting a healthier online ecology.

Related Works

Early text-based approach models used word embeddings such as Word2Vec to identify hateful keywords. This method achieved moderate success on HateSpeechDataset [8]. Later, transformer-based models were invented, and models such as BERT [9] significantly

improved this task by capturing relational information in context. BERT-based methods have achieved over 85% accuracy on text-only hate speech detection benchmarks. However, these models struggled with multimodal content where visual or auditory signals come into play. These additional modalities can complement or contradict textual cues, presenting a more complex task.

Image-based methods were used to address circumstances in which hate speech is communicated through visual content. CNN models can recognize hateful images with up to 82% accuracy on image-focused datasets such as HatefulMemes [10]. Unfortunately, these algorithms fail at handling textual or auditory context, which can lead to misclassification when nasty imagery is combined with benign content.

The combination of text and visual modalities has been critical in improving multimodal hateful content detection. The MMHS150K dataset [11] which contains 150000 tweets with matched text and images, has supported the construction of multimodal fusion networks. These models use pretrained models like BERT for text and ResNet for images. Their outputs are then combined using attention-based methods. Gómez et al. [11] discovered that fusion networks improve F1-scores by 10% compared to unimodal baselines, emphasizing the significance of cross-modal learning.

Building on these foundations, HateMM takes another huge step by adding audio signals with text and graphics. It features a dataset containing 43 hours of annotated video content. Das et al. [5] utilized this dataset to create deep learning models that analyze text with BERT, graphics with Vision Transformers (ViT), and audio using Mel-frequency cepstral coefficients (MFCC). Their multimodal fusion model combines features from all three modalities with cross-modal attention. This model outperformed text-only and image-only models by 5.7%, with an accuracy of 79.8% and a macro F1-score of 0.790.

The HateMM framework not only provides a strong baseline for multimodal hate speech detection, but it also emphasizes the importance of combining text, graphics, and audio for correct classification. Its emphasis on video content is consistent with the rising multimedia content available on internet video platforms. HateMM serves as the backbone of our study's exploration into MLLMs hateful video detection.

Method

Dataset

The core of this study is the HateMM dataset provided by the HateMM study [5]. This dataset consists of 43 hours of video content, all sourced from an unmoderated video platform called BitChute. All videos are in English, with some in other languages but with English subtitles. Each video is labelled with: hate or non_hate. The dataset also includes annotations for the target group and hate snippets, specifying the time spans of the hateful content within the video. For example, a video labeled as hateful might include a hate snippet annotation such as "00:01:30–00:01:45" with the note "targeted group: black," , meaning the content within this segment contains explicit hate symbols, offensive gestures, or derogatory speech targeting Black individuals. [5]

The dataset integrates three modalities. In our study, we treat each modality as follows:

- *Textual Data:* Initially, the transcript generated from the audio was considered as the textual modality. However, it became apparent that relying solely on the audio transcript could bypass audio analysis. Though it simplifies the process, it could lost potential insights. To investigate whether separate analysis of audio and text can enhance performance, we redefined the textual modality to include all directly displayed text, such as captions and subtitles, independent of the audio transcript.
- *Visual Data:* Video frames are processed to capture actions, gestures, and objects.
- *Auditory Data:* Audio tracks are analyzed to detect tonal cues, speech patterns, and ambient sounds.

Preprocessing

Baseline. Our baseline approach requires comprehensive preprocessing because it lacks pretrained encoders to directly process raw videoinputs. While MLLMs are equipped to extract hierarchical features and align modalities automatically, our baseline depends on manual feature extraction to transform video frames, tokenized text and converted

audio signals. In addition, preprocessing ensures alignment between modalities for computational efficiency. This step is essential since it enables effective multimodal analysis in the absense of pretrained encoders. We used Vosk offline speech recognition to generate video transcripts for the text modality, and extracted raw audio tracks from the videos. Visual data was processed by sampling 100 frames per video, ensuring uniform distribution throughout duration. Padding was performed for shorter videos to maintain a fixed number of frames for every video.

MLLMs. For multimodal large language models (MLLMs), the preprocessing step is reduced due to their ability to handle raw or minimally processed inputs. The preprocessing steps for the MLLMs used in this study are as follows:

- **Gemini-1.5-pro:** No preprocessing is required for this model. Raw video inputs are directly fed into the model using Google Cloud, as it is equipped to process the video data end-to-end.
- **LLaVA Next, LLaVA OneVision and VideoLLama2:** These models are designed for image-text modalities. We sample 30 frames from each video, uniformly distributed throughout its duration, and feed these frames directly into the models without additional preprocessing other than resizing.
- **VideoLLama2 Audio Visual:** This model supports video, audio, and text modalities simultaneously. We use the model processor to handle all modalities. The raw video, audio, and text inputs are passed through the processor which automatically extracts the necessary features for each modality and aligns them for downstream tasks.

Models

Baseline - HateMM.

Feature Engineering.

Text. After obtained from Vosk offline speech recognition tool, the transcripts are tokenized and fed to a pretrained BERT model. This gave us a contextual embeddings with dimension of 768×1. These embeddings are then passed through two fully connected layers with hidden dimensions of 128, followed by ReLU activation and batch

normalization for stability. Finally, a dense layer reduces the representation to a 64 $\times$ 1 feature vector, which encapsulates the semantic information from the transcripts.

Audio. We used Mel-frequency cepstral coefficients (MFCC) as the audio feature representations. Each audio sample is also padded to same duration for a consistent 40-dimensional shape. We also condense the temporal information into representative feature vectors by compuing the mean of the coefficients over time. After that, we pass these 40 x 1 feature vectors through two fully connected layers, each with 128 hidden units and ReLU activation, followed by batch normalization for stability. Finally, we obtain another 64 x 1 feature vector, but this time with the audio information.

Visual. Each extracted frame was divided into patches and went through Vision Transformer (ViT). ViT embeds these patches sequentially and then passes them through a series of transformer encoder layers, resulting in a 768 $\times$ 100 feature matrix for each video. This approach extracts complicated spatial relationships and contextual information from videos. This output feature matrix is then fed through an LSTM network, which handles the sequential nature of video frames. The LSTM layer creates a sequence representation, which is combined using an attention mechanism to focus on important frames. The output is processed through a dense layer which reduce the dimension to a 64 $\times$ 1. This final vector captures the most relevant spatial and temporal patterns for hate content detection.

Fusion and classification

The feature vectors from the three modalities, each with the shape of 64 x 1, are concatenated to form a fused feature vector of size 192 $\times$ 1. What happens here is basically aligning all the vectors horizontally to form a single vector. This combined vector will posses all information from text, audio and text. We then passed this fused feature vector through a fully connected fusion layer with ReLU activation to introduce non linearity and enhance the model's ability to learn complex relationships between modalities. The fusion layer reduces the dimensionality and prepares the integrated features for classification. The output is then fed into the final classification head, a dense layer with a softmax activation function, which computes a probability distribution over the binary classes (**hate** or **non-hate**). The softmax ensures that the predicted probabilities sum to 1, providing interpretable output and helps the model to make predictions about whether a video is hateful.

MLLMs.

Another big part of our study is that we made use of advanced MLLMs: open-source Gemini-1.5-pro as well as closed-source LLaVA-Next-Video, LLaVA-OneVision and VideoLlama2. LLaVA-Next and LLaVA-OneVision are specifically designed for image-text processing; meanwhile both Gemini-1.5-pro and VideoLlama2–audio-visual come with additional audio modality making them perfect for the video tasks. We sampled 30 frames for each video and used them as inputs for all models. We choose these models because thet are the most cutting-edge ones, as LLaVA-Next and LLaVA-OneVision excellent in image-text reasoning tasks, such as video captioning comprehension, and Gemini-1.5-pro and VideoLlama2-audio-visual are both good at audio-visual tasks like video classification and content moderation.

Experiments

We trained the baseline model on the HateMM dataset, split into training, validation and testing set with a ratio of 8:1:1, for 20 epochs and a learning rate of 1×10^{-4} with the Adam optimizer. We also employed a strategy called k-fold cross-validation to ensure the robustness of our results. On the ohter hand, the MLLMs were evaluated in their pretrained state without any additional finetuning. All models were assessed under consistent input conditions, and their performance was evaluated using accuracy, precision, recall, and F1 score.

For the baseline models, each modality was trained and evaluated separately. These feature vectors were then fed into neural network models trained using cross-entropy loss. Late fusion was performed in multimodal models by concatenating the outputs from each unimodality, followed by a classification head. In addition, improvement methods were investigated for potential performance gains. Accuracy and F1 scores were utilized to evaluate baseline models. We also tested on different modality combinations, from unimodality to combination of two, and then to all three combined. We aim to discover if this gradual addition of modality will result increase in performance.

MLLMs were tested using a standard setup of 30 frames per video. Audio inputs were only used in models that support audio-visual modalities (Gemini-1.5-pro and VideoLlama2-audio-visual). We used the general prompt: "Is there any hateful con-

tent in this video?" Answer 'Yes' or 'No' and explain why", to generate model response for all models. Positive and negative predictions were obtained by parsing the generated responses for the keywords 'yes' and 'no.' The results were then evaluated based on accuracy, precision, recall, and F1 scores.

Evaluation

Evaluation Metrics

The evaluation metrics used for both baseline and MLLMs are defined as follows:

- **Accuracy:**

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP (true positives) and TN (true negatives) are accurately predicted instances of "hate" and "non-hate", while FP (false positives) and FN (false negatives) are the inaccurate predictions.

- **Precision:**

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision is the proportion of accurately predicted "hate" instances among all that are predicted as "hate."

- **Recall:**

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall reflects the proportion of instances with true label 'hate' that were correctly identified by the model.

- **F1 Score:**

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1 Score is the harmonic mean of precision and recall.

- **Macro-F1 Score:**

$$\text{Macro-F1} = \frac{1}{C} \sum_{i=1}^{C} \text{F1 Score}_i$$

where C is the number of classes (in this case, two: "hate" and "non-hate"). This metric distributes equal weight to both groups, making it robust to class imbalance.

Results

The performance of the baseline models and MLLMs is summarized in Tables 1 and 2.

Table 1: Baseline Results (Accuracy and Macro-F1 Score)

Modality	Baseline	Reduce LR	Xavier Init.	Linear Complexity	Residual Blocks
MFCC	71.28 70.04	70.82 69.77	67.40 66.27	71.09 70.14	**71.38** **70.39**
ViT	70.91 69.41	70.91 69.41	**74.95** **72.67**	72.85 71.03	73.96 72.45
BERT	76.91 75.48	76.91 75.48	74.42 73.13	71.46 72.48	**74.24** **72.78**
BERT+MFCC	76.45 75.49	**78.48** **77.09**	78.48 77.02	77.47 76.32	78.12 76.77
BERT+ViT+MFCC	78.21 76.39	79.22 78.01	78.66 77.49	77.65 76.52	**79.78** **78.56**

Table 2: MLLM Results (Accuracy, Precision, Recall, and F1 Score)

Model	Accuracy	Precision	Recall	F1 Score
Gemini-1.5-pro	64.38%	42.74%	94.64%	58.89%
VideoLlama2	62.44%	54.81%	30.54%	39.22%
VideoLlama2-Audio Visual	47.16%	40.21%	75.62%	52.50%
LLaVA-Next-Video	55.86%	46.25%	67.28%	54.80%
LLaVA-OneVision	65.83%	80.19%	18.79%	30.45%

These results provide some unexpected findings. From the **Baseline** column in Table 1, it's clear that utilizing multi-modalities is consistently improving the performance of the model. However, combining text with audio (BERT + MFCC) yields no substantial gains in performance. The model with text input alone (BERT) can achieve 76.91% accuracy whereas the additional audio feature caused it's accuracy to drop for 0.5%. We speculate the model is only matching raw audio features to labels without using audio as a complementary source of information, specifically semantic information. As a result, the potential benefits of audio–text fusion remain largely unrealized. The second interesting finding is: despite the use of several improvement methods, the overall performance of the baseline model did not improve much. For example, including residual blocks in the baseline model only increased accuracy and F1 scores. We see the highest

state our model can achieve with improvement methods is with residual block, but still gaining only 1.5% in accuracy. The change isn't minor but we are still bound below the 80% threshold, meaning one out of every five videos will be falsely predicted and results in many hateful videos being overlooked. The advances did not produce a meaningful breakthrough we desire. This suggests that in order to improve the overall performance of hateful content detection in general, we need to review the feature extraction and fusion approaches rather than minor modifications to the current architecture.

For the pretrained MLLMs results in Table 2, the overall worse performances is quite shocking. Despite being trained on exponentially more data, none of these models outperform the baseline. It's understandable that LLaVA-OneVision outperformer LLaVA-Next-Video as the former is a more finetuned version. The close performance between Gemini-1.5-pro and LLaVA-OneVision also suggests that success in hateful video detection might not necessarily rely on using a larger or more complex model. However, VideoLlama2-AV's poor performance is the nost intriguing part. Gemini-1.5-Pro and VideoLlama2-AV are the only two models in our selection capable of processing audio, but neither of them can surpasses the baseline model's performance. After a closer investigation, we found out that Gemini-1.5-Pro extracts the transcripts from audio during processing with speech recognition systems [12]. Whereas VideoLlama2-AV process audio with an audio encoder that convert audio signals into mel spectrogram representation [13]. Both methods have its own strengths and limitations. Gemini-1.5-Pro benefits from speech recognition which integrates semantic meaning but may lose tonal audio features. VideoLlama2-AV reserve the raw audio features since it process the audio signals directly, but this will result in semantic meaning loss. We revisited the model responses and found VideoLlama2-AV has a tendency to identify unseen videos as hateful. This raises awareness of current audio processing approaches in MLLMs. Most audio encoders in MLLMs use processor similar to Mel Spectrograms encoders, which record the intensity and tone of audio signals but do not preserve semantic information, whereas hateful content is typically represented in subtle semantic distinctions in spoken language. Audio signals are simply not enough to win this task and can even act as noise during this task. Which is proven by the better performances of LLaVA-OneVision and LLaVA-Next-Video which omit the use of audio.

A possible reason both models fail to outperform the baseline lies in their insufficient

understanding of the alignment between modalities. Multimodal learning requires the model to establish meaningful correspondences between modalities. Gemini-1.5-Pro's reliance on text transcription might reduce the contextual relationship between audio and visual data. Similarly, VideoLlama2-AV's mel spectrogram-based audio processing fail to capture semantic relationships between audio, text, and video modalities. Another important conclusion we can think of, is that the baseline model's overall better performance is likely due to task-specific optimization. The baseline model's additional and precise training phase help to achieve a better modality alignment and feature integration ability. In comparison, Gemini-1.5-Pro and VideoLlama2-AV, both are general-purpose pretrained models, lack the fine-tuned alignment ability for this case. Research suggests that inadequate multimodal alignment is a significant barrier to improving performance in such models [14]. Moreover, pretrained models often suffer from modality collapse in which certain modalities dominate the learned representations, reducing cross-modal integration overall [15].

These findings indicate that we need better audio encoders that can bridge the gap between raw audio features and the semantic meaning obtained from audio transcripts. Potential methods include using pretrained speech recognition models to translate audio into semantically meaningful representations. Such as Wav2Vec2. Furthermore, combining these semantic-rich features with visual and text modalities may result in a more comprehensive approach to multimodal hate detection.

In conclusion, while baseline improvements and MLLMs have various degrees of success, results suggest that considerable advancements in hateful video identification need innovations in audio processing and alternative cross-modal fusion techniques. Solving these problems would not only increase MLLM performance, but also make it easier to detect subtle and hidden hateful text.

Data reevaluation

In order to validate our analysis from the test results, we performed a reevaluation of the HateMM dataset. The goal was to enhance the dataset with richer information that can help us to understand the factors behind MLLMs' underperformances. From the original HateMM dataset, we randomly selected 200 **hate** videos and re-annotated them based on the specific modalities contributing to the hateful content. For example, hate

videos that used audio and text to express their message were tagged with two Xs for audio and text modalities respectively. In addition, we completely reviewed all of the videos, providing detailed explanations for why each video is hateful. This included identifying specific audio segments, critical phrases in the text, and identifying objects that indicated as hateful. We believe that this expanded knowledge will help MLLMs improve their reasoning abilities during future fine-tuning. We also adjusted all hate-snippet annotations to reflect the correct spatial span of hateful content. Statistics from the re-annotated videos show that of the 200 hateful videos, 77 have hateful visual elements, 185 have hateful audio, and 67 have hateful text. We assessed the MLLMs on unimodality of these videos and found that all models performed well on the text modality alone, with accuracy ranging from 90% to 97%. However, audio modality performance was consistently poor across all models, backingup our conclusion on the significant weakness of current audio encoders in MLLMs. These audio encoders lose important semantic information. If we use an early-fusion strategy to align and concatenate modalities before feeding them into the model, allowing the model to use cross-modality interactions. May make sure that the semantic context provided by text and visual modalities supports audio perception and performance.

Conclusion

Based on the HateMM study, we took a step further by evaluating SOTA MLLMs for recognizing hateful videos. Even though the HateMM dataset provided useful insights into multimodality, certain MLLMs are still having trouble with capturing semantic complexities in spoken language. This is because modern audio encoders rely heavily on basic acoustic properties such as Mel Spectrograms. To investigate these gaps, we undertook a targeted re-annotation of chosen videos, adding more granular labels for hate snippets and tying them to probable hostile cues in text and images. This strategy identified instances where our models misread background audio cues that arrived intermittently.

In the future, we plan to implement an early-fusion method that aligns transcripts, audio tracks, and visual objects based on hate_snippet annotations. We will also utilize semantic audio encoders instead of intensity-only representations of audio, which preserves more information about spoken meaning and context. We want to create a more

complete detection method by synchronizing semantically driven audio embeddings with precisely calculated hate snippet timestamps and relevant video frames. This separates hateful elements as they appear. We believe that by focusing on alignment, we will be able to enhance overall classification accuracy while also enabling clearer interpretability and more efficient moderation procedures, so building on HateMM's strong foundation.

References

[1] Statista Research Department. Hours of video uploaded to youtube every minute as of 2023, 2023.

[2] Digital Marketing Experts. How much video content is uploaded daily across platforms?, 2023.

[3] YouTube Transparency Report. Algorithmic amplification of content, 2023.

[4] Yiheng Liu Chong Ma Xu Zhang Yi Pan Mengyuan Liu Peiran Gu Sichen Xia Wenjun Li Yutong Zhang Zihao Wu Zhengliang Liu Tianyang Zhong Bao Ge Tuo Zhang Ning Qiang Xintao Hu Xi Jiang Xin Zhang Wei Zhang Dinggang Shen Tianming Liu Shu Zhang Jiaqi Wang, Hanqi Jiang. A comprehensive review of multimodal large language models: Performance and challenges across different tasks. *arXiv:2408.01319*, 2024.

[5] Manisha Das, Arjun Patel, and Kavya Sharma. Hatemm: Multimodal hate speech detection in video content. *arXiv preprint arXiv:2305.03915*, 2023.

[6] Online Safety Institute. The evolving nature of hate speech online, 2023.

[7] R. Williams. Challenges in annotating multimodal hate speech datasets. *AI Ethics and Society*, 6(1):45–63, 2022.

[8] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. *ICWSM*, pages 512–515, 2017.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[10] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Michael Ringgaard, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *arXiv preprint arXiv:2005.04790*, 2020.

[11] Rodrigo Gomez and Vasilis Kalogeiton. Mmhs150k: A dataset for multimodal hate speech detection. *Proceedings of EMNLP*, pages 1230–1240, 2022.

[12] Google DeepMind. Gemini-1.5-pro documentation. *Google AI Documentation*, 2024.

[13] OpenAI. Videollama2-av pretrained model overview. 2024.

[14] J. Huang, A. Patel, and Y. Sun. Challenges in multimodal alignment for pretrained models. *Journal of Multimodal Research*, 15(4):455–467, 2023.

[15] Z. Wu and L. He. Modality collapse in multimodal neural networks: Causes and solutions. In *Proceedings of NeurIPS*, volume 134, pages 2231–2242, 2022.