Summary of Changes for Resubmission

OpenReview URL of Previous KDD Submission: https://openreview.net/forum?id=7gERKJeovf
Title: MatchLM2Lite: A Scalable MLLM-to-Lite Framework for Reproduced Content Identification

Dear Program Committee,
We thank the reviewers for their time and constructive feedback. The feedback has been valuable for clarifying key points and improving our work. We summarize all improvements and clarifications for our resubmission below:

Major Changes

- **Addressed Visual Illustration Clarity for Multimodality Improvements:** We revised the original Figure 4 in our paper with clearer captions and expanded it to show the score distributions of cases with positive RCI label for modality-specific subsets and the full test set. We included a subset relationship diagram for better clarity. We updated this figure to more clearly demonstrate the performance gains when incorporating additional modalities into **MatchLite**, and show specifically that including additional modalities improves the predictive power of the model.
- **Included Comprehensive Knowledge Distillation Ablation Results:** We added Table 5 & Table 6, which contain additional ablation experiments for different Knowledge Distillation hyperparameters and further analyze the effects of different Knowledge Distillation settings in Section B.3.
- **Addressed NTP Experiment AP Results:** We updated the NTP results in Table 3 using the prediction logits softmaxed over the given options in our multiple choice task to compute AP and F1 and updated the implementation details of this in section B.1 accordingly.
- **Added Standard Deviation Measures as Indicator of Result Stability:** We carried out additional experiments with different random seeds and added standard deviation estimates for **MatchLite**'s (with Knowledge Distillation) and **MatchLM**'s AP and F1 under section 4.3, exemplifying the stability of our results.
- **Clarified Variations in Visual-only Baseline across Dataset Subsets:** We clarified that different subsets of our test-set whose labels are influenced specifically by different audio/text reasons were isolated from the full test-set. This was done intentionally to illustrate that the addition of relevant modalities can help improve MatchLite's predictive power relative to the visual-only baseline in each of these cases. As these are subsets with different labelling reasons, they naturally give rise to slightly different but comparable visual-only baseline values.
- **Supplement Dataset Details:** We updated section 4.1 with size of dataset and percentage of reproduced video pairs.
- **Supplement Model Details:** We updated Appendix D.1 and section D.2 with learning rate, optimizer and scheduler details.
- **Addressed Generalization Concerns on In-House Datasets:** We updated section 4.4 with details on performance of Match2Lite on other in-house video deduplication and live-stream deduplication datasets, exemplifying the generalizability of the improvements from our Match2Lite System across multiple datasets.
- **Supplemented Online Deployment Details:** We updated section 5.1 with details on the latency of our upstream retrieval pipeline, MatchLite latency and the entire end-to-end pipeline latency, as well as the average and peak QPS that our systems handles.

Minor Corrections

- We added a clear y-axis to Figure 5 and clarified the description and motives for experimental results.
- We fixed a small error in Table 2: MatchLM+'s F1 score is 79.12, not 78.53. The performance increment stated in 4.3 was updated accordingly.
- We added an additional baseline performance using video copy segment localization model TransVCL to Table 1.
- We rearranged Related Works Section 2 to before our methodology and experiments to improve readability and coherence.
- We revised the manuscript to address grammatical issues and refine the writing, thereby improving clarity and flow.
- We simplified the results in Table 2 by combining results from Data Size with the results from additional In-house Task Pretraining into a single section.
- We reordered the sections in Section 4: Experiments for better readability and shift the training details to the Appendix D.
- We made minor edits in our figures for greater clarity and neatness.

MatchLM2Lite: A Scalable MLLM-to-Lite Framework for Reproduced Content Identification

Xiaotian Fan*
TikTok, Singapore
xiaotian.fan@tiktok.com

Hiok Hian Ong*
TikTok, Singapore
hiok.ong@tiktok.com

David Yuchen Wang*
TikTok, Singapore
david.w@tiktok.com

Zirui Zhu
TikTok, Singapore
zirui.zhu@tiktok.com

Kanchan Sarkar†
TikTok, San Jose, USA
kanchan.sarkar@tiktok.com

Kun Xu†
TikTok, San Jose, USA
daniel.chen28@tiktok.com

Abstract

Content moderation is critical for online video platforms to ensure content safety, protect creators, and sustain positive user experiences. Beyond filtering harmful content, platforms must guarantee content authenticity at scale so that users are exposed to diverse, original videos rather than low-value reproductions. We present **MatchLM2Lite**, a real-time, production-grade reproduced content identification (RCI) system that leverages the powerful understanding of a multimodal large language model (MLLM) distilled into a small and fast-inference model. Our system jointly models video, audio, and text signals, operating on pairs of videos to produce fine-grained reproduction scores. The system comprises two modules, **MatchLM** and **MatchLite**, and a two-stage training recipe. First, our high-capacity MLLM, MatchLM, serves as a teacher model to define the upper bound of RCI performance. Its capabilities are then distilled into a compact student model, MatchLite. This design allows MatchLite to deliver low-latency, high-throughput inference on video pairs while preserving much of MatchLM's accuracy, making it suitable for integration into real-time recommendation systems. MatchLM achieves an F1-score improvement of **+8.57** compared to our previous production model. After knowledge distillation, MatchLite retains a **+6.55** gain in F1-score while reducing computational cost by **35x**. Deployed at scale, MatchLM2Lite enables efficient, pairwise multimodal RCI, stably serving online traffic at high queries per second (QPS) with an end-to-end latency below 30 seconds. This system has reduced the reproduced video view rate on our platform by **2.5%** without degrading user engagement, demonstrating its effectiveness in a large-scale production environment.

CCS Concepts

• **Information systems** → **Recommender systems**; • **Computing methodologies** → **Artificial intelligence**; **Language models**.

Keywords

Multimodal Large Language Models, Content Intelligence, Reproduced Content Identification, Video Understanding, Knowledge Distillation, Scalable Inference

ACM Reference Format:
Xiaotian Fan, Hiok Hian Ong, David Yuchen Wang, Zirui Zhu, Kanchan Sarkar, and Kun Xu. 2026. MatchLM2Lite: A Scalable MLLM-to-Lite Framework for Reproduced Content Identification. In *Proceedings of Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (Conference acronym KDD)*. ACM, New York, NY, USA, 14 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Content moderation has become a crucial concern for ensuring video quality and user experiences on short video platforms such as Instagram Reels, Pinterest, TikTok, and Kuaishou, to provide healthy and creative content while reducing harmful and low-quality content [30, 54].

Content authenticity protection is a key part of content moderation, aiming to reduce the spread of copied or minimally modified videos. These efforts protect the rights of content creators, enhance content diversity, improve recommendation quality, and contribute to a more sustainable and safe ecosystem.

The core technical challenge is reproduced content identification (RCI) which refers to the identification of variants modified by trimming, filtering, or editing between query and candidate videos. It requires joint modeling of paired videos across multiple modalities to capture detailed similarities and overall semantic information. Existing methods mainly rely on embedding-based similarity using visual encoders [12, 13, 18, 19], ignoring key text and audio signals which users engage with when watching short videos.

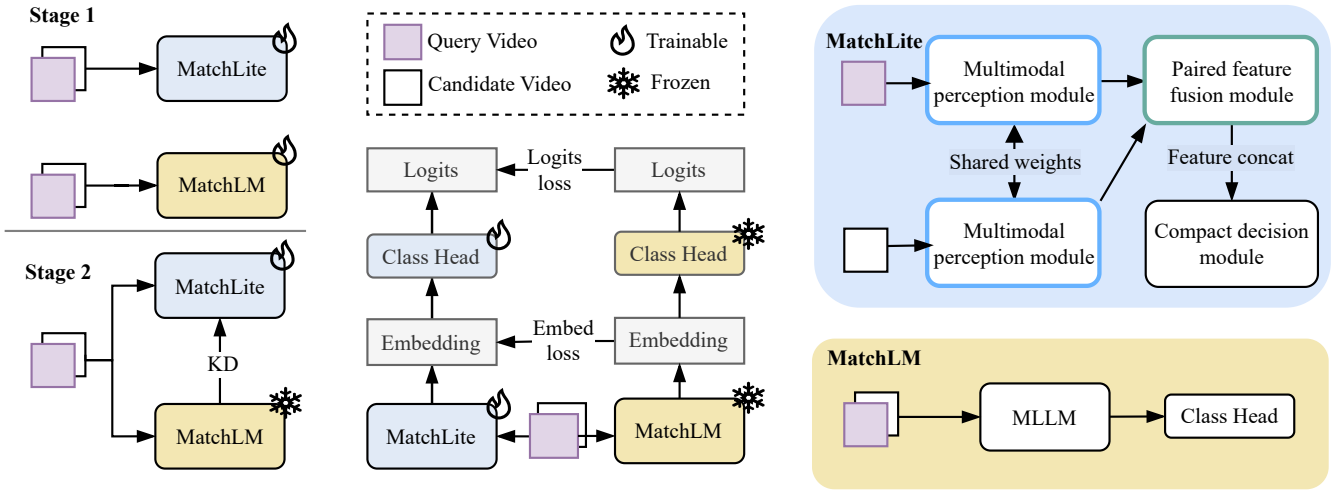
To address these limitations, we propose a **MatchLM2Lite** framework consisting of a high-capacity multimodal large language model **MatchLM**, for setting a strong reference point for model performance, along with a lightweight multimodal model **MatchLite**, for real-time serving. Both models jointly encode visual, audio and text, enabling stronger cross-modal interactions.

We train this framework in a two-stage approach. In the first stage, **MatchLM** and **MatchLite** are trained in a supervised manner

*Equal contribution.
†Project lead.

Conference acronym KDD, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2026/06
https://doi.org/XXXXXXX.XXXXXXX



(a). MatchLM2Lite framework (b). Knowledge distillation (c). Workflow of MatchLM & MatchLite

Figure 1: Overall design of MatchLM2Lite (a). MatchLM2Lite Framework involves two stages of training: Stage 1 - Teacher & student trained separately with GT labels. Stage 2 - Freeze teacher; train student with supervised + distillation losses; (b). Knowledge Distillation. Student learns from frozen MLLM via cosine embedding loss and KL divergence on logits, along with standard cross-entropy loss; (c). Workflow of MatchLM & MatchLite.

using labeled data. In the second stage, **MatchLM** is frozen and acts as a "teacher" to distill its knowledge into **MatchLite** [53]. This distillation process enables **MatchLite** to inherit the capabilities of **MatchLM** with strong cross-modal alignment and rich semantic representation, while maintaining its compact architecture suitable for large-scale production deployment.

We validate the proposed **MatchLM2Lite** framework through extensive offline experiments and online A/B testing on a large-scale, real-world short video platform. It shows strong effectiveness in production content governance workflows, with full-scale deployment supporting a throughput of over 3k requests per second.

Our key contributions are as follows:

- We introduce a unified multimodal framework for reproduced content identification, formulating and modeling it as a video pair matching problem across three modalities: visual, audio, and text. We apply joint encoding and alignment of multimodal signals to enable robust reproduced content identification.
- We design a **MatchLM2Lite** system that consists of a MLLM-based teacher model (**MatchLM**) with a lightweight student model (**MatchLite**) for efficient real-time deployment. **MatchLM** learns rich cross-modal representations for classification, while **MatchLite** maintains semantic alignment and efficiency through knowledge distillation.
- We deploy our approach at scale on our short video platform and achieve consistent improvements in both offline and online settings. Compared to our prior production model, **MatchLM** achieves an F1 improvement of +8.57; **MatchLite** achieves an F1 of +6.55 after knowledge distillation. Online A/B tests show a 2.5% reduction in reproduced content views, demonstrating production effectiveness.

2 Related Works

2.1 Content Moderation

Content moderation is crucial for shielding users from harmful content and protecting creators from plagiarism [2, 36, 37]. Human-based content moderation demand significant human labor [6], can cause emotional distress through exposure to large amounts of toxic/harmful content [43], and is susceptible to biases affecting fairness and consistency [32]. Model-driven moderation can alleviate both economic and psychological costs of human moderation while enhancing the safety and quality of online content. Previous works have employed Neural Networks for toxic-content detection [10, 21, 42, 46], as well as localization modules for video-copy detection [12, 13, 29]. Online systems typically engage large-scale retrieval systems such as [9], which are used to find candidate matches for a given query. This generates paired samples which then need to be evaluated for content duplication [13], reproduction, or other policy-dependent violations.

2.2 Multi-Modal Interaction with LLMs

The development of multi-modal large language models (MLLMs)[1, 22, 24, 25, 50] have employed various alignment techniques to integrate both visual and textual inputs, thereby enhancing video comprehension in conjunction with language. LLaVA-One-Vision [22] is among one of the top performing open-source MLLMs which leverages the pre-trained Qwen-2 [55] language backbone and SigLIP [56] vision encoder. However, most existing powerful MLLMs are pretrained on only one or a limited set of modalities, such as language, or vision-language. Meanwhile, existing small audio models such as Whisper [38] and related works [4, 33, 34, 47, 49] focus narrowly on specific audio domains like human speech or natural sounds [7]. Efforts have aimed to extend MLLMs to handle diverse

audio inputs, by injecting audio information into pretrained backbones. QwenAudio [5] integrates diverse audio signals through an early-fusion approach to encode audio embeddings into a large language model. AudioPaLM [40] utilizes weights from a text-based model and fuses this with audio-based models to improve speech processing. Video-LLaMA [58] utilizes separate branches for different modalities and leverages existing pretrained embeddings from ImageBind [11] to learn visual-audio-language correspondence. Qwen-Omni [51] [52] similarly integrates audio, visual and text modalities and trains an end-to-end MLLM to achieve holistic perception capabilities. These works demonstrate success in leveraging pretrained large vision-language model weights to improve performance in additional modalities.

2.3 Industrial Applications

Recent works have explored the potential of MLLMs for tasks such as recommendation and content moderation. For example, NoteLLM2 [57] proposes joint end-to-end finetuning of existing LLMs and vision encoders to create strong multi-modal embeddings of visual and text content, which is utilized in their content recommendation systems. Similarity, QARM [31] extends this approach by incorporating audio information into their MLLM, further enhancing recommendation capabilities. Kuaishou also demonstrated that vision-language models can serve as effective online content moderators [30], highlighting the versatitilty of MLLMs in diverse business applications.

2.4 Knowledge Distillation

One disadvantage of using MLLMs is their large size and high latency which is heavily resource consuming. Knowledge distillation [15] can distill the knowledge from large, cumbersome models down smaller sized models more suited to production environments. Previous works have shown that large LLM/MLLMs can serve as effective teachers to teach both smaller-scale LLM/MLLMs [3, 44, 45], or other small models [20], achieving performance surpassing the student model itself. The predicted target distributions from the teacher can be directly used as a loss objective for the predicted distributions for the student. [41].

3 Methodology

The main challenge of reproduced content identification lies in detecting a variety of modifications between paired videos while avoiding penalizing the non-copied videos (false positives). Existing solutions for reproduced video detection typically utilize a visual encoder and focus only on matching visual similarities [12, 13]. As RCI is an internal task based on specific policies, no similar public benchmarks exist. Some previous in-house solutions for this task propose multi-tower retrieval frameworks that separate each modality into standalone recall pipelines with late-stage fusion modules. However, this design leads to greater system complexity, higher serving latency, and also weakens cross-modal alignment.

3.1 MatchLM2Lite Training Framework

To address these limitations, we propose the **MatchLM2Lite** framework to balance accuracy and efficiency. In this section, we introduce the architectural design of **MatchLM** and **MatchLite** and the end-to-end two stage training recipe for the framework, as illustrated in Figure 1 (a). In first stage training, both **MatchLM** and

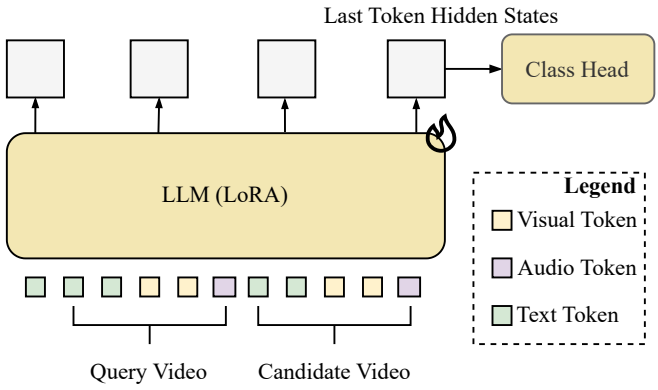


Figure 2: MatchLM architecture design. Query and candidate video embeddings are interleaved and fed into the LLM. The final hidden state of the last token serves as the video representation, followed by a classification head for prediction.

MatchLite are trained for paired video matching across modalities and are independently finetuned with supervised ground-truth labels. **MatchLM** adopts an early-fusion strategy by transforming visual frames, audio, and accompanying text into token sequences and uses multimodal projectors for modality alignment. The token sequences are then processed by an LLM backbone fine-tuned via LoRA [16]. **MatchLite** uses efficient multimodal encoders to extract and fuse modality-specific features, as shown in Figure 1 (c).

In the second stage, knowledge distillation is performed to transfer the learned multimodal semantic representations of **MatchLM** to **MatchLite**. Specifically, **MatchLM** parameters are frozen, while **MatchLite** is further trained using a combination of distillation loss and supervised classification loss, leveraging the pretrained checkpoints from stage one, as shown in Figure 1 (b). More details are provided in section 3.4.2.

3.2 MatchLM Architecture Design

We build **MatchLM** upon LLaVA-One-Vision (0.5B) (LLaVA-OV) [22]: the architecture incorporates SigLIP as the visual encoder and leverages Qwen2 as the LLM backbone. We extend the model to simultaneously process audio [5, 58].

We begin with separate encoding strategies for each modality to obtain their respective token representations. For the visual and text modalities, we adopt the default LLaVA-One-Vision [22] pipeline, where video frames are sampled at fixed intervals, encoded via SigLip-So400m-Patch14-384 [56], and injected into the LLM through the original vision-to-language projector.

To incorporate audio, inspired by QwenAudio [5], we use the Whisper-small [38] encoder to extract high-level representations from raw audio. The input audio is sampled at 16kHz and transformed into 80-dimensional log-mel features, from which we extract 1500 audio tokens per video. A special `<audio>` token is inserted to denote the position of the audio embeddings in the input sequence. Additionally, a learnable audio saliency weighting layer is used to

further aggregate the sequence of audio tokens into a single audio token. Instead of using Q-Former [23, 27, 58] as a modality bridge like in Video-LLaMA [58], we follow the same design in LLaVA-One-Vision to use a lightweight MLP projector [22, 27] to map audio tokens to the same embedding dimension as text and visual input embeddings. The final input to the LLM consists of interleaved tokens from all modalities, beginning with task-specific prompts. To enable pairwise video comparison, we extend the standard MLLM input format to support two-video inputs—a query video and a candidate video—within a single forward pass, as shown in Figure 2. The input sequence is constructed as:

$$\texttt{Prompt} + [\texttt{Video}_1] + [\texttt{Audio}_1] + [\texttt{Video}_2] + [\texttt{Audio}_2].$$

We utilize the MLLM model as a paired video representation extractor [26, 59] rather than as a next-token prediction generative model. This design enables direct use of rich semantic embeddings for discriminative classification and helps effectively distill knowledge into a lightweight **MatchLite** model. Specifically, we use the final hidden state corresponding to the last token as the paired video representation [8]. On top of the LLM output, we append a lightweight classification head. The representation is passed through a shared projection layer followed by task-specific output layers.

This structure supports various content reproduction classification tasks while maintaining low inference cost. We use cross-entropy loss for each task and aggregate them as the final multi-task classification objective:

$$\mathcal{L}_{\text{class}} = \sum_{t=1}^{T} \mathcal{L}_{\text{CE}}^{(t)} \tag{1}$$

In practice, this consists of the main RCI task prediction layer alongside other auxiliary task prediction layers for sublabels (e.g. subtitles) which are discarded at inference time.

3.3 MatchLite Architecture Design

Next, we focus on **MatchLite**, the lightweight student model designed for efficient deployment. As shown in Figure 1 (c), **MatchLite** consists of three main modules: Multimodal Perception, Paired Feature Fusion, and a Compact Decision Module.

The Multimodal Perception Module utilizes pretrained, frozen encoders—Swin Transformer [28] for vision, Sentence-BERT [48] for text, and Whisper-small [38] for audio. Modality-specific features f for each video V_x ($x \in \{q, c\}$), are extracted yielding f_x^v (visual), f_x^t (text), and f_x^a (audio), where q denotes query and c denotes candidate. To further enhance visual representations, we employ a modality mutual injection mechanism based on bidirectional cross-attention (BiXT) [14], which produces four mixed-modality embeddings for each video:

$$[f_x^{vt}, f_x^{tv}] = \text{BiXT}^{(v\leftrightarrow t)}(f_x^v, f_x^t),$$
$$[f_x^{va}, f_x^{av}] = \text{BiXT}^{(v\leftrightarrow a)}(f_x^v, f_x^a), \tag{2}$$

where $f_x^{vt}, f_x^{tv}, f_x^{va}$, and f_x^{av} denote the mixed-modality embeddings for video V_x. The process is shown in Figure 3 (a).

The Paired Feature Fusion Module jointly processes the features of the query and candidate videos. For each modality $m \in v, va, vt, a, av, t, tv$, we collect the corresponding pair of features $[f_q^m, f_c^m]$. For visual-related modalities ($m \in v, va, vt$), a dense pairwise tensor

$$D_{i,j}^m = \text{Concat}(f_{q,i}^m, f_{c,j}^m)$$

is constructed and passed through several CNN layers to obtain enhanced pairwise features E^m. In parallel, cosine similarity matrices

$$S_{i,j}^m = \text{cosine}(f_{q,i}^m, f_{c,j}^m)$$

are computed for all modalities. All enhanced feature maps E^m and similarity matrices S^m are concatenated as $\mathcal{F}$ for downstream processing. The overall procedure is summarized in Algorithm 1.

All fused features $\mathcal{F}$ are passed to a lightweight ResNet-34-based Compact Decision Module, which produces the final multimodal representation for classification. A multi-task classification head further supports multiple downstream tasks, shown in Figure 3 (b).

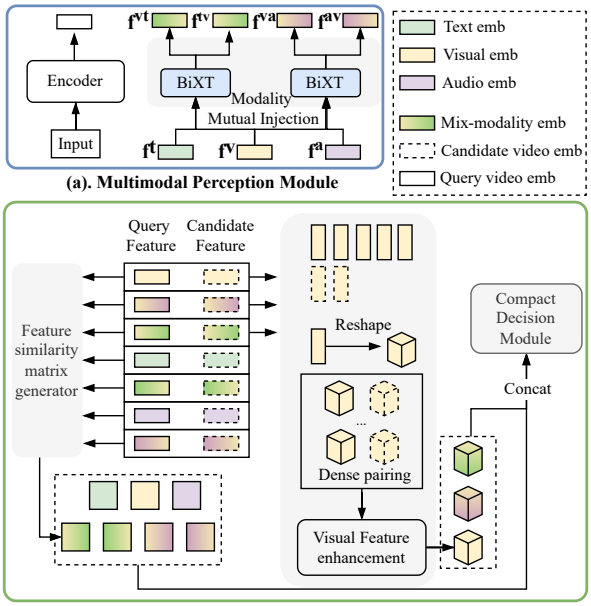


Figure 3: Main modules of MatchLite. (a). Multimodal Perception Module: Frozen encoders extract modality-specific features. BiXT applies bidirectional cross-attention between modality pairs; (b). Paired Feature Fusion Module: Computes cosine similarity between query-candidate feature pairs. Visual features are further fused via lightweight CNNs. All outputs are concatenated for final decision.

3.4 Two-Stage Training Recipe

3.4.1 Stage 1: Supervised training. We train the **MatchLM** and **MatchLite** independently using standard supervised training for paired video classification using the same training data and task objectives, as shown in Figure 1 (a). The supervision loss is formulated as the sum of cross-entropy losses across all downstream classification tasks, shown as Equation 1.

3.4.2 Stage 2: Knowledge distillation. Although **MatchLM** achieves strong performance, deploying it at scale will incur significant GPU overhead. Therefore, to improve efficiency during deployment, we

Algorithm 1: MatchLite reproduced content identification

Input: Query video V_q, Candidate video V_c
Output: Match score s

// **Multimodal Perception**
Initialize $\mathcal{O} = \emptyset$;
for *each* $x \in \{q, c\}$ **do**
  $f_x^v \leftarrow \text{Encoder}^{(v)}(V_x^v)$;
  $f_x^t \leftarrow \text{Encoder}^{(t)}(V_x^t)$;
  $f_x^a \leftarrow \text{Encoder}^{(a)}(V_x^a)$;
  $[f_x^{vt}, f_x^{tv}] \leftarrow \text{BiXT}^{(v\leftrightarrow t)}(f_x^v, f_x^t)$;
  $[f_x^{va}, f_x^{av}] \leftarrow \text{BiXT}^{(v\leftrightarrow a)}(f_x^v, f_x^a)$;
for *each modality* $m \in \{v, va, vt, a, av, t, tv\}$ **do**
  $o^m = [f_q^m, f_c^m]$;
  Append o^m to $\mathcal{O}$;
// **Paired Feature Fusion**
Initialize $\mathcal{E} = \emptyset$, $\mathcal{S} = \emptyset$;
for *each* $o^m \in \mathcal{O}$ **do**
  $[f_q^m, f_c^m] = o^m$;
  if $m \in \{v, va, vt\}$ **then**
    $D_{i,j}^m = \text{Concat}\left(f_{q,i}^m, f_{c,j}^m\right),\ i \in [1, M],\ j \in [1, N]$;
    $E^m = \text{ResBlockConv}(D^m)$;
    Append E^m to $\mathcal{E}$;
  $S_{i,j}^m = \text{CosineMap}(f_{q,i}^m, f_{c,j}^m)$;
  Append S^m to $\mathcal{S}$;
$\mathcal{F} = \text{Concat}(\mathcal{E}, \mathcal{S})$;
// **Compact Decision**
$z = \text{ResNet34}(\mathcal{F})$;
$s = \text{MultiTaskHead}(z)$;
return s;

adopt a knowledge distillation strategy **MatchLM2Lite** to transfer its capabilities to **MatchLite**, as illustrated in Figure 1 (b). We apply both embedding distillation and logit distillation.

For embedding-level distillation, we extract the final hidden state of the last input token from **MatchLM** as a unified paired video-level representation as well as the last embedding state from **MatchLite**. A learnable projection head is applied to the MLLM outputs to transform it into the dimension of the student embeddings. We then minimize the cosine distance between the student representation $\mathbf{z}_s$ and the teacher representation $\mathbf{z}_t$:

$$\mathcal{L}_{\text{emb}} = 1 - \cos(\mathbf{z}_s, \mathbf{z}_t) = 1 - \frac{\mathbf{z}_s \cdot \mathbf{z}_t}{|\mathbf{z}_s||\mathbf{z}_t|} \tag{3}$$

For logit distillation, we apply a Kullback–Leibler (KL) divergence loss to align the predictive distributions between **MatchLM** and **MatchLite**, on the softmax of their predicted logits.

$$\mathcal{L}_{\text{logits}} = \text{KL}\left(\sigma\left(\frac{\mathbf{p}_t}{T}\right) \middle| \sigma\left(\frac{\mathbf{p}_s}{T}\right)\right) \tag{4}$$

Here, $\sigma(\cdot)$ denotes the softmax function.

Our two-part distillation loss combines both embedding-level and logit-level training objectives and is given by:

$$\mathcal{L}_{\text{distill}} = \mathcal{L}_{\text{emb}} + \mathcal{L}_{\text{logits}} \tag{5}$$

Finally, the overall training objective linearly combines the distillation loss with the task-specific classification loss:

$$\mathcal{L}_{\text{total}} = \lambda \cdot \mathcal{L}_{\text{distill}} + \mathcal{L}_{\text{class}} \tag{6}$$

We set the weighting coefficient $\lambda = 1.5$ to strengthen the influence of distillation versus classification during training.

4 Experiments

4.1 Dataset

We construct a multimodal, video-paired Reproduced Content Identification (RCI) dataset using videos from our platform. For this, we leverage an in-house recall pipeline based on visual features. We first sample a group of query videos, and recall the top 1 candidate video based on visual similarity for each query to form a query/candidate pair. These pairs are sent for human-labeling to obtain their final RCI label. In total, we obtain 0.8 million video pairs with 4.77% being reproduced pairs.

To ensure comprehensive coverage, we curate our dataset by generating query videos by 1) random sampling daily published videos, which represent the common types of normal/reproduced content, and 2) sampling from high engagement videos, which are daily videos that have gone viral or pseudo-viral and thus garnered higher popularity, which represent specialized cases of reproduced content. This ensures that the dataset distribution accounts for both the large majority of normal/reproduced content types as well as those that with greater risk of being reproduced due to their high user viewership.

We treat RCI as a binary classification task, and focus on the following metrics: F1-score, Average Precision (AP), and recall at precision 80% (R@P80). R@P80 is used as an important metric to ascertain the expected proportion of reproduced content that can be recalled after the model is deployed online. Our end goal is to identify reproduced content with high precision and apply suppression strategies on copied video content while also ensuring minimal overkill (false positive) cases for the full platform traffic.

4.2 Preliminary Exploration

We perform preliminary explorations to test the zero-shot capabilies of existing open/close sourced models on our RCI task. We first investigate powerful MLLMs including GPT-4o [35], Qwen2.5VL [1] and LLaVA-OV [22], as well as the open-source TransVCL model [12], which is one of the best models on the video-copy detection. Despite these MLLMs having strong general multimodal capabilities and TransVCL being pretrained on large-scale video copy-localization data, all the models yielded relatively poor performance, as shown in Table 1. Even when provided with the policy and task description, zero-shot models are unable to reliably capture our content-type-specific governance policy or adapt to fast-changing content trends on the platform. These results motivate the need for supervised finetuning on our in-house RCI training set.

Table 1: Zero-shot evaluations on our RCI testset using existing models. For TransVCL, cls: uses the cls-token with a linear layer as a classifier and localization: uses the detected bounding box directly from TransVCL to assess whether the video contains a duplicated segment.

Model Name	Open sourced	Modalities	**F1**
TransVCL (cls)	Y	V	14.84
TransVCL (localization)	Y	V	15.17
GPT-4o	N	V+T	22.69
Qwen2.5VL 3B	Y	V+T	8.6
LLaVA-OV 0.5B	Y	V+T	12.2
+ SFT on in-house data	Y	V+T+A	77.25

Note: For GPT-4o we use model version gpt-4o-2024-11-20

4.3 Main Experiment Results

The results of our model training on our RCI dataset are presented in Table 2. As a baseline, we train a visual-only version of **MatchLite**, without the use of the Multimodal Perception Module. We find that incorporating audio and text modalities significantly improves upon the visual only baseline model. We further perform data scaling experiments to dive into the sample efficiency of the models and show that in-house pretraining also boosts model performance. Finally, by integrating Knowledge Distillation, we bridge the gap in performance between the teacher **MatchLM** and the student **MatchLite**. With all techniques applied, the final performances for **MatchLite** is AP=82.45±0.09, F1=77.10±0.08 and for **MatchLM** is AP = 86.21±0.078, F1=79.12±0.30 with standard deviation estimates from 3 experimental runs with additional random seeds. The teacher model, **MatchLM** outperforms the **MatchLite** in AP (+3.76) and F1 (+2.02). Our training hyperparameter details can be found in Appendix D.

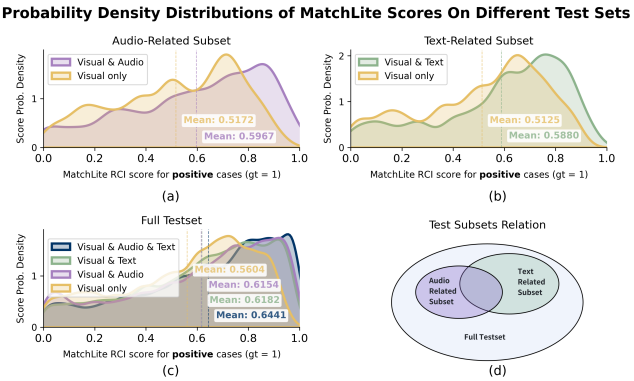


Figure 4: Different subsets are isolated from the full testset to observe the performance gains due to the addition of modalities. (a) For a subset whose labels are influenced by audio, adding audio improves performance. (b) For a subset whose labels are influenced by text, adding text improves performance. (c) For the full test set, combining audio and text yields the largest gains, while either modality alone still outperforms the visual-only model. (d) Relationships among the test subsets.

4.3.1 Additional Modality Studies. We investigate the benefits of incorporating additional modalities and compare the performance of **MatchLite** with 4 variants: 1) Baseline (Visual only **MatchLite**) 2) Visual + Audio **MatchLite** 3) Visual + Text **MatchLite** 4) Visual + Text + Audio **MatchLite**. As seen from Table 2, the inclusion of any of the additional modalities like audio and text will improve the performance of **MatchLite**, with the inclusion of the audio modality accounting for comparatively larger additional improvement (+3.39 AP/+3.26 F1) than the inclusion of the text modality (+0.83 AP/+0.94 F1). Finally, adding all 3 modalities will yield the best improvement of +3.68 AP/+3.56 F1. Similarly, for **MatchLM** we observe a performance boost when incorporating audio modality during training, accounting for an additional improvement of +3.99 AP/+2.27 F1.

To further analyze the impact of adding modalities, we extract targeted subsets from the main test set and visualize the score distributions for each model. As shown in Figure 4, adding audio shifts the score distribution in the positive cases audio-reason subset toward higher reproduction scores, and adding text produces a similar shift in the positive cases text-reason subset. Adding modalities also shifts the distribution for the positive cases in the full test set in the correct direction.

4.3.2 Data Scaling. Data scaling experiments were conducted on **MatchLite** and **MatchLM** as shown in Table 2. **MatchLM** significantly outperforms **MatchLite** with only 1/3 of the training data, demonstrating that **MatchLM** is more sample efficient under data-constrained conditions. More details are provided in Appendix B.4. Additionally, we leverage pretrained model checkpoints trained on other in-house content moderation tasks into both **MatchLite** and **MatchLM**'s training and find an additional +2.21 AP/+0.37 F1 for **MatchLite** and an additional +1.22 AP/+0.59 F1 for **MatchLM**.

4.3.3 Knowledge distillation. Resource constraints present a real challenge when it comes to serving the full real-time traffic on our video sharing platform. While **MatchLM** achieves the best performance, it is significantly more resource intensive than **MatchLite**, posing an important question of whether the return on investment is justified when deploying this model for our full daily traffic volume. Taking this into consideration, we instead apply knowledge distillation using **MatchLM** as the teacher model and the **MatchLite** as the student model.

Concretely, we perform knowledge distillation from MatchLM to MatchLite via KL divergence on classification logits and cosine distance loss on embeddings, both with $\lambda = 1.5$, yielding a +9.71% recall at P80 as seen in Table 4. From our ablation studies, we find that the benefits of knowledge distillation plateau at $\lambda = 1.5$. Increasing further to $\lambda = 2$ yielded comparable results. Finally we selected the model trained with $\lambda = 1.5$ for deployment as it achieved the highest increase in AP (+1.86) and R@P80 (+9.71), with its F1 marginally trailing the $\lambda = 2$ config by 0.2. Detailed ablations and analysis on varying $\mathcal{L}_{emb}$ and $\mathcal{L}_{logits}$ can be found in Appendix B.3, along with visualisations of differences the learned features maps in MatchLite with and without distillation.

A performance gap remains: the **MatchLM** achieves R@P80 of 75.92, with 9% higher recall than the distilled **MatchLite**. We identify two possible reasons: (1) the feature backbones of the **MatchLite** are frozen during serving to support shared feature

Table 2: Experiment results on input modality, training data size, and knowledge distillation. We compare the results for MatchLite and MatchLM across the integration of different modalities (visual/audio/text) and perform experiments using different data scales. Finally, we show the improvements in performance when MatchLite is distilled from MatchLM. The baseline model configuration is underlined, the best MatchLite configuration is bolded and '+' refers to the addition of in-house task pretraining dataset

Setting	Model	Modality	Data (%)	AP	F1
Modality	Baseline	V	100	74.70	70.55
	MatchLite	V + T		75.53 (+0.83)	71.49 (+0.94)
		V + A		78.09 (+3.39)	73.81 (+3.26)
		V + A + T		78.38 (+3.68)	74.11 (+3.56)
	MatchLM	V + T	100	81.00	76.26
		V + A + T		84.99 (+3.99)	78.53 (+2.27)
Data Size	MatchLite	V + A + T	33	77.08	72.84
			66	78.14 (+1.06)	73.68 (+0.84)
			100	78.38 (+1.30)	74.11 (+1.27)
	MatchLite+	V + A + T	100+	80.59 (+3.51)	74.48 (+1.64)
	MatchLM	V + A + T	33	82.63	77.53
			66	83.15 (+0.52)	77.63 (+0.10)
			100	84.99 (+2.36)	78.53 (+1.00)
	MatchLM+	V + A + T	100+	86.21 (+3.58)	79.12 (+1.59)
Distillation	MatchLite+ (w/o KD)	V + A + T	100+	80.59	74.48
	MatchLite+ (w KD)*			**82.45** (+1.86)	**77.10** (+2.62)
	MatchLM+			86.21	79.12

Table 3: MatchLM Ablation Studies: We compare the effects of using Last Token Classification with Next Token Prediction (NTP), Early audio fusion with Late audio fusion, Dynamic vs Static Frame Allocation and the difference in performance when using Qwen2.5VL 3B backbone vs LLaVA-OV 0.5B.

MatchLM Model	Objective	Audio Fusion	Frame Allocation	AP	F1
Qwen2.5VL 3B	Last Token Cls	Early	Dynamic	83.38	77.45
LLaVA-OV 0.5B	NTP	Early	Dynamic	83.13	77.81
LLaVA-OV 0.5B	Last Token Cls	Late	Dynamic	83.51	77.65
LLaVA-OV 0.5B	Last Token Cls	Early	Static	84.83	77.86
LLaVA-OV 0.5B	Last Token Cls	Early	Dynamic	**84.99**	**78.53**

Table 4: Ablation study on Knowledge Distillation on the best performing MatchLite+ (with the addition of in-house task pretraining dataset): The effect of varying $\mathcal{L}_{\text{logits}}$ and $\mathcal{L}_{\text{emb}}$ shows that performance improves and then plateaus at $\mathcal{L}_{\text{logits}}$=1.5 and $\mathcal{L}_{\text{logits}}$=1.5.

Model	$\mathcal{L}_{\text{logits}}$	$\mathcal{L}_{\text{emb}}$	AP	F1	R@P80
MatchLite+	–	–	80.59	74.48	57.21
MatchLite+	1.0	1.0	81.95	76.75	64.26
MatchLite+	1.5	1.5	**82.45**	77.10	**66.92** (+9.71)
MatchLite+	2.0	2.0	82.37	77.29	65.70

+ add In-house Task Pretraining

caching and minimize inference cost, which limits its ability to learn new patterns effectively; and (2) **MatchLite** has significantly fewer parameters than **MatchLM**, leading to lower representation capacity even with Knowledge Distillation.

4.3.4 MatchLM Ablation Studies. To study the contribution of individual components in **MatchLM**, we conduct several ablation experiments: (1) next-token prediction (NTP) formulated as a binary multiple-choice task (reproduced vs. not reproduced) versus last-token classification with a dedicated head; (2) early versus late audio fusion; (3) dynamic versus static frame allocation (with padding to 60 seconds); and (4) replacing the MLLM backbone (Qwen2.5-VL 3B) with LLaVA-OV 0.5B under the best-performing configuration. Results are summarized in Table 3.

We adapt our task to the NTP paradigm by framing it as a multiple-choice question as a baseline for comparison with our last-token classification objective. Last-token classification consistently outperforms NTP, likely because it constrains the prediction space to two classes rather than the full vocabulary. More prompt details are provided in Appendix B.1.

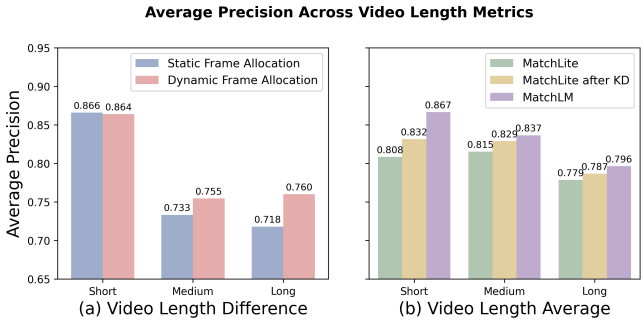


Figure 5: (a) Effectiveness of Dynamic Sampling versus Static Frame Allocation in MatchLM under different video length differences. Dynamic Sampling adapts more effectively to mismatched video lengths, and its effects amplify as the length disparity between the video pair grows. (b) Performance of models across varying average video lengths. The capabilities of all models tend to drop as the average length of video pairs increase, while KD proves effective across all video lengths.

We further compare early and late audio fusion strategies. For late fusion, we mean-pool audio embeddings and concatenate them with the MLLM's last-token embedding before classification. For early fusion, the aggregated audio token is interleaved directly into the input token sequence. Additional implementation details are included in Appendix A.2 and Appendix B.2. Early fusion outperforms late fusion, achieving gains of +1.48 AP and +0.88 F1. This improvement likely arises because early fusion enables richer cross-modal interactions among audio, visual, and textual tokens, whereas late fusion relies on a limited linear combination and cannot fully exploit the MLLM's reasoning capacity.

Furthermore, we observe that dynamic frame allocation—assigning frames to each video proportionally to their length—outperforms static allocation in both AP (+0.16) and F1 (+0.67). As shown in Figure 5(a), dynamic sampling is particularly beneficial when video pairs exhibit large length discrepancies, as it preserves more contextual information. Sampling details are provided in Appendix A.2. Finally, applying the best configuration to the Qwen2.5-VL 3B backbone yields comparable but slightly lower performance than LLaVA-OV 0.5B (Table 3).

4.4 Cross-Domain Generalization

We further extend the task to carry out reproduced content identification on whether a published video is reproduced from source content such as movies, TV shows, or dramas using **MatchLM** and achieve a significant improvement over its single-video input MLLM (mono-model), increasing R@P80 from 58.72 to 86.5. More details on this task, termed as Hierarchical Reproduced Content Identification (H-RCI), can be found in Appendix C.

To validate the generalization capability of the **MatchLM2Lite** framework across domains and data scales, we evaluate the model using additional in-house deduplication datasets. On a short video deduplication task, P80 recall improves from 74.7% to 81.8%, and on a live streaming deduplication task, P80 recall improves from 49.2% to 68.8%, demonstrating the robustness of the proposed approach.

5 Online Serving and Experiments

5.1 Online Serving

MatchLite requires only 0.86 TFLOPs per inference, compared to 25.4 TFLOPs for **MatchLM**, yielding a 35x reduction in serving-time computational cost on an NVIDIA A10 24GB GPU. Leveraging knowledge distillation, we preserve **MatchLite**'s efficiency while achieving a 9.71% improvement in R@P80 under the same computational budget.

We integrate the best **MatchLite** model into our online systems for RCI detection. When a video is published, an upstream retrieval module first selects the top-1 candidate from a vector database based on visual similarity. The original and retrieved videos are then passed to **MatchLite**, which produces an RCI score used by the recommendation system to prioritize original content over reproduced ones. All newly published videos on the platform are scored in real time, and those with high RCI scores are deboosted. In our production deployment, retrieval latency is approximately 12s, **MatchLite** inference latency is about 2.8s, and the end-to-end pipeline latency remains below 30s. The overall system stays stable under an average load of 2.8k queries per second (QPS), with peak QPS exceeding 3.5k.

5.2 Online Experiments

We conducted a two-week online A/B experiment on our short video platform, allocating 10% of the overall online video traffic across 2 groups. Users were randomly assigned to either a control or treatment group. RCI scores used for moderation in the control group was determined by a visual-only online baseline model, while the treatment group deployed the distilled **MatchLite** to identify reproduced content. We evaluated the impact on two key business metrics: stay duration (the average time users spend on the platform, measuring user engagement) and reproduced video views (the number of viewed videos that are reproduced). **MatchLite** reduced the reproduced-video view rate by 2.5% without significantly affecting user stay duration. Throughout the A/B experiment, we monitored core user metrics and platform health indicators to assess both effectiveness and potential side effects. After post-experiment review, we deployed **MatchLite** to serve full traffic.

6 Conclusion and Future Work

In this work, we propose a **MatchLM2Lite** framework for reproduced content identification (RCI) in our short video platform. Our approach utilizes a powerfull MLLM, **MatchLM** to provide a strong modality alignment and rich semantic representation. Later, we distill its knowledge to **MatchLite** for low-latency and high-throughput in production scenarios while retaining performance. This framework demonstrates a practical and scalable solution for large-scale industry deployment, while also opening up new directions for multimodal video copy detection using MLLMs.

For future work, we will further explore the capabilities of the MLLM model by involving user feedback for continual learning. We also plan to extend our approach to longer video sequences through token merging or compression strategies. Beyond classification of reproduced content, we aim to leverage the MLLM model for temporal grounding of copied video segments, enabling more fine-grained and explainable content detection.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. 2025. Qwen2.5-VL Technical Report. arXiv:2502.13923 [cs.CV] https://arxiv.org/abs/2502.13923

[2] Akash Bonagiri, Lucen Li, Rajvardhan Oak, Zeerak Babar, Magdalena Wojcieszak, and Anshuman Chhabra. 2025. Towards Safer Social Media Platforms: Scalable and Performant Few-Shot Harmful Content Moderation Using Large Language Models. arXiv:2501.13976 [cs.CL] https://arxiv.org/abs/2501.13976

[3] Yuxuan Cai, Jiangning Zhang, Haoyang He, Xinwei He, Ao Tong, Zhenye Gan, Chengjie Wang, Zhucun Xue, Yong Liu, and Xiang Bai. 2025. LLaVA-KD: A Framework of Distilling Multimodal Large Language Models. arXiv:2410.16236 [cs.CV] https://arxiv.org/abs/2410.16236

[4] Yi-Chen Chen, Po-Han Chi, Shu wen Yang, Kai-Wei Chang, Jheng hao Lin, Sung-Feng Huang, Da-Rong Liu, Chi-Liang Liu, Cheng-Kuang Lee, and Hung yi Lee. 2021. SpeechNet: A Universal Modularized Model for Speech Processing Tasks. arXiv:2105.03070 [cs.CL] https://arxiv.org/abs/2105.03070

[5] Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-Audio: Advancing Universal Audio Understanding via Unified Large-Scale Audio-Language Models. arXiv:2311.07919 [eess.AS] https://arxiv.org/abs/2311.07919

[6] Dan Cosley, Dan Frankowski, Sara Kiesler, Loren Terveen, and John Riedl. 2005. How oversight improves member-maintained communities. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Portland, Oregon, USA) *(CHI '05)*. Association for Computing Machinery, New York, NY, USA, 11–20. doi:10.1145/1054972.1054975

[7] Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. 2024. Pengi: An Audio Language Model for Audio Tasks. arXiv:2305.11834 [eess.AS] https://arxiv.org/abs/2305.11834

[8] Xin Dong, Sen Jia, Ming Rui Wang, Yan Li, Zhenheng Yang, Bingfeng Deng, and Hongyu Xiong. 2025. COEF-VQ: Cost-Efficient Video Quality Understanding through a Cascaded Multimodal LLM Framework. arXiv:2412.10435 [cs.CV] https://arxiv.org/abs/2412.10435

[9] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The Faiss library. arXiv:2401.08281 [cs.LG] https://arxiv.org/abs/2401.08281

[10] Akshat Gaurav, {Brij B.} Gupta, {Kwok Tai} Chui, Varsha Arya, and Priyanka Chaurasia. 2023. Deep Learning Based Hate Speech Detection on Twitter. In *2023 IEEE 13th International Conference on Consumer Electronics - Berlin, ICCE-Berlin 2023 (IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin)*. doi:10.1109/ICCE-Berlin58801.2023.10375620 Publisher Copyright: © 2023 IEEE.; 13th IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin 2023 ; Conference date: 04-09-2022 Through 05-09-2022.

[11] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. ImageBind: One Embedding Space To Bind Them All. arXiv:2305.05665 [cs.CV] https://arxiv.org/abs/2305.05665

[12] Sifeng He, Yue He, Minlong Lu, Chen Jiang, Xudong Yang, Feng Qian, Xiaobo Zhang, Lei Yang, and Jiandong Zhang. 2022. TransVCL: Attention-enhanced Video Copy Localization Network with Flexible Supervision. arXiv:2211.13090 [cs.CV] https://arxiv.org/abs/2211.13090

[13] Sifeng He, Xudong Yang, Chen Jiang, Gang Liang, Wei Zhang, Tan Pan, Qing Wang, Furong Xu, Chunguang Li, Jingxiong Liu, Hui Xu, Kaiming Huang, Yuan Cheng, Feng Qian, Xiaobo Zhang, and Lei Yang. 2022. A Large-scale Comprehensive Dataset and Copy-overlap Aware Evaluation Protocol for Segment-level Video Copy Detection. arXiv:2203.02654 [cs.CV] https://arxiv.org/abs/2203.02654

[14] Markus Hiller, Krista A. Ehinger, and Tom Drummond. 2024. Perceiving Longer Sequences With Bi-Directional Cross-Attention Transformers. arXiv:2402.12138 [cs.CV] https://arxiv.org/abs/2402.12138

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the Knowledge in a Neural Network. arXiv:1503.02531 [stat.ML] https://arxiv.org/abs/1503.02531

[16] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685 [cs.CL] https://arxiv.org/abs/2106.09685

[17] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nZeVKeeFYf9

[18] Qing-Yuan Jiang, Yi He, Gen Li, Jian Lin, Lei Li, and Wu-Jun Li. 2019. SVD: A Large-Scale Short Video Dataset for Near-Duplicate Video Retrieval. 5280–5288. doi:10.1109/ICCV.2019.00538

[19] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. 2017. Near-Duplicate Video Retrieval with Deep Metric Learning. doi:10.1109/ICCVW.2017.49

[20] Taegyeong Lee, Jinsik Bang, Soyeong Kwon, and Taehwan Kim. 2025. Multi-aspect Knowledge Distillation with Large Language Model. arXiv:2501.13341 [cs.CV] https://arxiv.org/abs/2501.13341

[21] Alyssa Lees, Vinh Q. Tran, Yi Tay, Jeffrey Sorensen, Jai Gupta, Donald Metzler, and Lucy Vasserman. 2022. A New Generation of Perspective API: Efficient Multilingual Character-level Transformers. arXiv:2202.11176 [cs.CL] https://arxiv.org/abs/2202.11176

[22] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024. LLaVA-OneVision: Easy Visual Task Transfer. arXiv:2408.03326 [cs.CV] https://arxiv.org/abs/2408.03326

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. arXiv:2301.12597 [cs.CV] https://arxiv.org/abs/2301.12597

[24] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. arXiv:2311.10122 [cs.CV] https://arxiv.org/abs/2311.10122

[25] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. 2024. VILA: On Pre-training for Visual Language Models. arXiv:2312.07533 [cs.CV] https://arxiv.org/abs/2312.07533

[26] Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. 2025. MM-Embed: Universal Multimodal Retrieval with Multimodal LLMs. arXiv:2411.02571 [cs.CL] https://arxiv.org/abs/2411.02571

[27] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual Instruction Tuning. arXiv:2304.08485 [cs.CV] https://arxiv.org/abs/2304.08485

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. arXiv:2103.14030 [cs.CV] https://arxiv.org/abs/2103.14030

[29] Minlong Lu, Yichen Lu, Siwei Nie, Xudong Yang, and Xiaobo Zhang. 2025. Self-supervised Video Copy Localization with Regional Token Representation. In *Computer Vision – ECCV 2024*, Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (Eds.). Springer Nature Switzerland, Cham, 18–35.

[30] Xingyu Lu, Tianke Zhang, Chang Meng, Xiaobei Wang, Jinpeng Wang, YiFan Zhang, Shisong Tang, Changyi Liu, Haojie Ding, Kaiyu Jiang, Kaiyu Tang, Bin Wen, Hai-Tao Zheng, Fan Yang, Tingting Gao, Di Zhang, and Kun Gai. 2025. VLM as Policy: Common-Law Content Moderation Framework for Short Video Platform. arXiv:2504.14904 [cs.SI] https://arxiv.org/abs/2504.14904

[31] Xinchen Luo, Jiangxia Cao, Tianyu Sun, Jinkai Yu, Rui Huang, Wei Yuan, Hezheng Lin, Yichen Zheng, Shiyao Wang, Qigen Hu, Changqing Qiu, Jiaqi Zhang, Xu Zhang, Zhiheng Yan, Jingming Zhang, Simin Zhang, Mingxing Wen, Zhaojie Liu, Kun Gai, and Guorui Zhou. 2024. QARM: Quantitative Alignment Multi-Modal Recommendation at Kuaishou. arXiv:2411.11739 [cs.IR] https://arxiv.org/abs/2411.11739

[32] Henrietta Lyons, Eduardo Velloso, and Tim Miller. 2021. Conceptualising Contestability: Perspectives on Contesting Algorithmic Decisions. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (April 2021), 1–25. doi:10.1145/3449180

[33] Soumi Maiti, Yifan Peng, Shukjae Choi, Jee weon Jung, Xuankai Chang, and Shinji Watanabe. 2024. Voxtlm: unified decoder-only models for consolidating speech recognition/synthesis and speech/text continuation tasks. arXiv:2309.07937 [eess.AS] https://arxiv.org/abs/2309.07937

[34] Eliya Nachmani, Alon Levkovitch, Roy Hirsch, Julian Salazar, Chulayuth Asawaroengchai, Soroosh Mariooryad, Ehud Rivlin, RJ Skerry-Ryan, and Michelle Tadmor Ramanovich. 2024. Spoken Question Answering and Speech Continuation Using Spectrogram-Powered LLM. arXiv:2305.15255 [cs.CL] https://arxiv.org/abs/2305.15255

[35] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug

Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. GPT-4o System Card. arXiv:2410.21276 [cs.CL] https://arxiv.org/abs/2410.21276

[36] Konstantina Palla, José Luis Redondo García, Claudia Hauff, Francesco Fabbri, Henrik Lindström, Daniel R. Taber, Andreas Damianou, and Mounia Lalmas. 2025. Policy-as-Prompt: Rethinking Content Moderation in the Age of Large Language Models. arXiv:2502.18695 [cs.CY] https://arxiv.org/abs/2502.18695

[37] Ed Pizzi, Giorgos Kordopatis-Zilos, Hiral Patel, Gheorghe Postelnicu, Sugosh Nagavara Ravindra, Akshay Gupta, Symeon Papadopoulos, Giorgos Tolias, and Matthijs Douze. 2023. The 2023 Video Similarity Dataset and Challenge. arXiv:2306.09489 [cs.CV] https://arxiv.org/abs/2306.09489

[38] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust Speech Recognition via Large-Scale Weak Supervision. arXiv:2212.04356 [eess.AS] https://arxiv.org/abs/2212.04356

[39] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20, Tutorial)*.

[40] Paul K. Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, Hannah Muckenhirn, Dirk Padfield, James Qin, Danny Rozenberg, Tara Sainath, Johan Schalkwyk, Matt Sharifi, Michelle Tadmor Ramanovich, Marco Tagliasacchi, Alexandru Tudor, Mihajlo Velimirović, Damien Vincent, Jiahui Yu, Yongqiang Wang, Vicky Zayats, Neil Zeghidour, Yu Zhang, Zhishuai Zhang, Lukas Zilka, and Christian Frank. 2023. AudioPaLM: A Large Language Model That Can Speak and Listen. arXiv:2306.12925 [cs.CL] https://arxiv.org/abs/2306.12925

[41] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108 [cs.CL] https://arxiv.org/abs/1910.01108

[42] Diptanu Sarkar, Marcos Zampieri, Tharindu Ranasinghe, and Alexander Ororbia. 2021. FBERT: A Neural Transformer for Identifying Offensive Content. arXiv:2109.05074 [cs.CL] https://arxiv.org/abs/2109.05074

[43] Joseph Seering, Tao Wang, Joon Yoon, and Geoff Kaufman. 2019. Moderator engagement and community development in the age of algorithms. *New Media & Society* 21, 7 (2019), 1417–1443. doi:10.1177/1461444818821316

[44] Fangxun Shu, Yue Liao, Le Zhuo, Chenning Xu, Lei Zhang, Guanghao Zhang, Haonan Shi, Long Chen, Tao Zhong, Wanggui He, Siming Fu, Haoyuan Li, Bolin Li, Zhelun Yu, Si Liu, Hongsheng Li, and Hao Jiang. 2024. LLaVA-MoD: Making LLaVA Tiny via MoE Knowledge Distillation. arXiv:2408.15881 [cs.CV] https://arxiv.org/abs/2408.15881

[45] Yijun Tian, Yikun Han, Xiusi Chen, Wei Wang, and Nitesh V. Chawla. 2024. Beyond Answers: Transferring Reasoning Capabilities to Smaller LLMs Using Multi-Teacher Knowledge Distillation. arXiv:2402.04616 [cs.CL] https://arxiv.org/abs/2402.04616

[46] Ciprian-Octavian Truică, Ana-Teodora Constantinescu, and Elena-Simona Apostol. 2024. StopHC: A Harmful Content Detection and Mitigation Architecture for Social Media Platforms. arXiv:2411.06138 [cs.SI] https://arxiv.org/abs/2411.06138

[47] Tianrui Wang, Long Zhou, Ziqiang Zhang, Yu Wu, Shujie Liu, Yashesh Gaur, Zhuo Chen, Jinyu Li, and Furu Wei. 2023. VioLA: Unified Codec Language Models for Speech Recognition, Synthesis, and Translation. arXiv:2305.16107 [cs.CL] https://arxiv.org/abs/2305.16107

[48] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. arXiv:2002.10957 [cs.CL]

[49] Xiaofei Wang, Manthan Thakker, Zhuo Chen, Naoyuki Kanda, Sefik Emre Eskimez, Sanyuan Chen, Min Tang, Shujie Liu, Jinyu Li, and Takuya Yoshioka. 2024. SpeechX: Neural Codec Language Model as a Versatile Speech Transformer. arXiv:2308.06873 [eess.AS] https://arxiv.org/abs/2308.06873

[50] Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, Min Dou, Kai Chen, Wenhai Wang, Yu Qiao, Yali Wang, and Limin Wang. 2025. InternVideo2.5: Empowering Video MLLMs with Long and Rich Context Modeling. arXiv:2501.12386 [cs.CV] https://arxiv.org/abs/2501.12386

[51] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. 2025. Qwen2.5-Omni Technical Report. arXiv:2503.20215 [cs.CL] https://arxiv.org/abs/2503.20215

[52] Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. 2025. Qwen3-Omni Technical Report. *arXiv preprint arXiv:2509.17765* (2025).

[53] Shilin Xu, Xiangtai Li, Haobo Yuan, Lu Qi, Yunhai Tong, and Ming-Hsuan Yang. 2024. LLAVADI: What Matters For Multimodal Large Language Models Distillation. arXiv:2407.19409 [cs.CL] https://arxiv.org/abs/2407.19409

[54] Rintaro Yanagi, Yamato Okamoto, Shuhei Yokoo, and Shin'ichi Satoh. 2024. The Effects of Short Video-Sharing Services on Video Copy Detection. arXiv:2403.18158 [cs.CV] https://arxiv.org/abs/2403.18158

[55] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 Technical Report. arXiv:2407.10671 [cs.CL] https://arxiv.org/abs/2407.10671

[56] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid Loss for Language Image Pre-Training. arXiv:2303.15343 [cs.CV] https://arxiv.org/abs/2303.15343

[57] Chao Zhang, Haoxin Zhang, Shiwei Wu, Di Wu, Tong Xu, Xiangyu Zhao, Yan Gao, Yao Hu, and Enhong Chen. 2025. NoteLLM-2: Multimodal Large Representation Models for Recommendation. arXiv:2405.16789 [cs.IR] https://arxiv.org/abs/2405.16789

[58] Hang Zhang, Xin Li, and Lidong Bing. 2023. Video-LLaMA: An Instruction-tuned Audio-Visual Language Model for Video Understanding. arXiv:2306.02858 [cs.CL] https://arxiv.org/abs/2306.02858

[59] Xin Zhang, Yanzhao Zhang, Wen Xie, Mingxin Li, Ziqi Dai, Dingkun Long, Pengjun Xie, Meishan Zhang, Wenjie Li, and Min Zhang. 2025. GME: Improving Universal Multimodal Retrieval by Multimodal LLMs. arXiv:2412.16855 [cs.CL] https://arxiv.org/abs/2412.16855

A Architecture Details

A.1 MatchLite Details

Implementation Details. For visual and audio modalities, we extract the original video frame/audio binaries of the original video sampled at 1 FPS. For text modality, we extract user generated text for each video. Concretely, for each video, we dynamically sample up to the maximum frames that we can accomodate from the video and carry out zero padding for shorter videos. Each frame is resized to 224 x 224 and 128 dimension visual embeddings are extracted from each video using an in-house pretrained Swin-T visual encoder. An in-house pretrained Whisper-small encoder is used to extract 1500 x 768-dimension audio embeddings and mean pooled to 1 x 768 dimension to increase throughput during deployment. Finally, for text modality, we concatenate the video title and the sticker text overlay, and use an open source Sentence-BERT Multilingual-MiniLM-L12-H384 model [48] to extract 64 x 384-dimension embedding vector from each video text. Max length truncation to 64 tokens is used to minimise feature extraction latency. To account for the diverse variety of languages used on our video sharing platform, we chose a multilingual text encoder so as to have wider coverage and understanding of a variety of languages. Our architecture features an 8-head BiXT Multimodal Perception Module with hidden size of 64, a Visual Feature Enhancement Module comprising 3 Conv2D layers with BatchNorm, ReLU, and residual connections and a ResNet-34 with the first Conv2D layer resized so that the number of input channels correspond to the number of feature channels in our enhanced visual features.

A.2 MatchLM Details

Dynamic Frame Allocation. To provide sufficient contextual information, we dynamically allocate frames from the video pair for the input tokens to MatchLM based on video length ratios. This ensures a minimum of number of frames will be included for the shorter video, which may be fewer if the video is shorter than the minimum allocation. We then uniformly sample frames across both videos to capture temporal context up to a fixed total budget of frames across the 2 videos. Finally, we prefix each video's frames with "Video 1:" and "Video 2:" to distinguish them for the model.

Audio Token Saliency Weighted Aggregation. The audio modality is also interleaved within the model input, with a simple text description prefix "Audio: " before the audio of each video. We use a Whisper-small encoder to extract 1500 x 768 dimension audio embeddings. To further reduce the number of input tokens into **MatchLM**, we use a linear layer to learn the saliency scores for each audio token. These saliency scores are then softmaxed and used to compute a weighted sum of the audio tokens. Finally, this audio token is projected into the same embedding dimension as the text tokens before feeding them into the LLM model.

B Ablation details

B.1 Reframing Reproduced Content Identification as Next Token Prediction

In order to adapt our reproduced identification task to the Next Token Prediction (NTP) format, we frame the problem as a multiple choice task. We appended "Choose the correct option from the given options: A) Normal B) Reproduced. Answer:" and format the labels such that we train the model to predict A for normal content, and B for reproduced content. During validation, we take the last token logits for "A" and "B" and apply softmax, taking these as the prediction probabilities for the 2 options. This simple baseline is provided to compare between the popular NTP training objective and our last token prediction objective.

B.2 Late Audio Fusion

For late audio fusion, we fuse the mean pooled 1x768 audio embeddings of each video by concatenating them along the hidden dimension and passing it through a linear layer. This is then concatenated with the MLLM's last token embedding and fed into the classification head. The same cross-entropy loss as in the early audio fusion configuration is used to train the Last Audio Fused MatchLM.

B.3 Effect of Knowledge Distillation

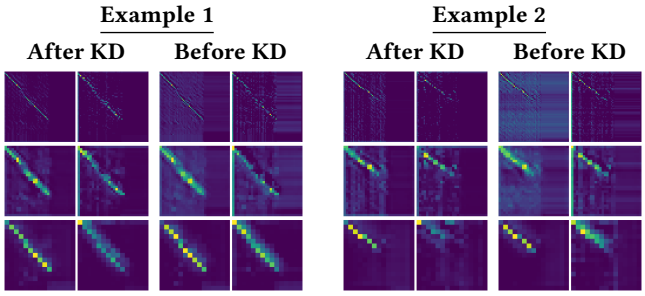


Figure 6: Feature maps after Knowledge Distillation qualitatively appear to be slightly more distinct, ignoring background noise features

We include detailed Knowledge Distillation ablations in Table 5 and Table 6. We find that incorporating KL Loss alone with a weight of $\mathcal{L}_{logits}$=1.5 and already improves F1 and AP significantly as shown in Table 5, raising F1 to 77.30% (+2.82%), AP to 82.26% (+1.67%) and R@P80 to 65.34% (+8.13%). Incorporating the Embedding Loss alone with a weight of $\mathcal{L}_{emb}$=1.5 also improves performance as shown in Table 6, raising F1 to 76.03% (+1.55%), AP to 81.65% (+1.06%) and R@P80 to 62.66% (+5.45%) but the improvement is not as large as $\mathcal{L}_{logits}$. This shows that both losses are useful in improving performance, with $\mathcal{L}_{logits}$ having a stronger effect, given its direct influence over the prediction logits. From Table 5, we see that as $\mathcal{L}_{logits}$ is increased, holding $\mathcal{L}_{emb}$ constant at 1.5, there is a steady improvement in the model's performance, while from Table 6 as $\mathcal{L}_{emb}$ is increased, holding $\mathcal{L}_{logits}$ constant at 1.5, we see a general increase in model performance as well especially in terms of R@P80, with slight fluctuations. The best setting is achieved when both losses are applied at $\mathcal{L}_{emb}$=1.5 and $\mathcal{L}_{logits}$=1.5.

Qualitatively, the intermediate feature maps learned by our Compact Decision module of **MatchLite** exhibits relatively clearer learned representations for reproduced video pairs after Knowledge Distillation as can be seen from Figure 6. This indicates that

Model	$\mathcal{L}_{\text{logits}}$	$\mathcal{L}_{\text{emb}}$	AP	F1	R@P80
MatchLite+	–	–	80.59	74.48	57.21
MatchLite+	–	1.5	81.65	76.03	62.66
MatchLite+	0.5	1.5	82.51	76.51	65.57
MatchLite+	1.0	1.5	82.37	76.54	65.02
MatchLite+	1.5	1.5	82.45	77.10	66.92

Table 5: Ablation study on Knowledge Distillation: Varying coefficient $\mathcal{L}_{\text{logits}}$ while keeping coefficient $\mathcal{L}_{\text{emb}}$ constant at 1.5

Model	$\mathcal{L}_{\text{logits}}$	$\mathcal{L}_{\text{emb}}$	AP	F1	R@P80
MatchLite+	–	–	80.59	74.48	57.21
MatchLite+	1.5	–	82.26	77.30	65.34
MatchLite+	1.5	0.5	81.90	76.25	66.59
MatchLite+	1.5	1.0	82.46	76.83	66.24
MatchLite+	1.5	1.5	82.45	77.10	66.92

Table 6: Ablation study on Knowledge Distillation: Varying coefficient $\mathcal{L}_{\text{emb}}$ while keeping coefficient $\mathcal{L}_{\text{logits}}$ constant at 1.5

MatchLM guidance might aid in learning better and more representative feature patterns that occur in the pairwise features.

From analyzing the test-set across average video lengths, we also find that knowledge distillation helps to improve the **MatchLite**'s performance across all 3 video length groups small, medium and long average videos as can be seen from Figure 5 (b).

B.4 Data Scaling Ablation Details

To investigate how **MatchLite** and **MatchLM**'s performance scales with the amount of data, we run a simple set of data scaling experiments to observe the effects of additional training data on model performance comparing the **MatchLite** model and the **MatchLM** model. As can be seen from Table 2, We find that even with only 1/3 the amount of training data, **MatchLM** with AP 82.63 and F1 77.53 already significantly outperforms **MatchLite** at AP 78.38 and F1 74.11 given the full amount of training data, demonstrating that **MatchLM** is more sample efficient and learns robust representations even under data-constrained conditions.

C Hierarchical Reproduced Content Identification: H-RCI

In addition to identifying reproduced content through paired video comparison, we extend the task to also use a single-video input to determine whether a published video is reproduced from source content such as movies, TV shows, or dramas. This setting extends the problem from pairwise comparison to a video understanding task. We name the new task as Hierarchical Reproduced Content Identification (H-RCI).

To address the broader H-RCI task, we propose a joint **MatchLM** model, built upon the original RCI **MatchLM** architecture. This unified model is capable of handling both paired and single video inputs within a single MLLM model. Evaluation results demonstrate that the joint model significantly improves overall H-RCI performance by effectively leveraging the contextual information provided by paired video inputs.

Table 7: Joint model on H-RCI task

Model	AP	F1	R@P80
Mono Model	72.93	67.73	58.72
Mono + Paired model	-	80.5	81.02
Joint Model	89.2	83.12	86.5

As shown in Table 7, when applying the single-video input MLLM model (mono MLLM) to the H-RCI task, it achieves an R@P80 of 58.72. As a comparison, we evaluate the usage of the mono model together with paired-video input MLLM (paired model) as a combined setup which improves the R@P80 to 81.02. Finally, our joint model, which integrates the paired video input as a pre-filtering mechanism for the video understanding branch, further increases R@P80 to 86.5 on the H-RCI task.

D Training Details

D.1 MatchLite Model

The **MatchLite** model comprises of 263 Million parameters including the frozen feature backbones and was trained on 16 NVIDIA A100 80GB GPUs with a batch size of 16 per GPU. AdamW optimizer with cosine annealing and a learning rate of 1e-4 was used and **MatchLite** was trained for 8 epochs. All three feature extractors are frozen for faster inference and resource saving through feature caching during deployment, with other layer weights are updated during training.

D.2 MatchLM Model

To manage memory during training, for our **MatchLM** model, we 1) adopted the relatively lightweight LLaVA-OV 0.5B model to reduce model size, 2) used dynamic frame allocation to accommodate varied video lengths, 3) early audio fusion using a learned saliency-based pooling layer. The **MatchLM** model consists of 1B parameters and is trained on 48 A100 GPUs with a batch size of 1 per GPU. AdamW optimizer with linear annealing and a learning rate of 2e-5 was used and **MatchLM** was trained for 8 epochs. For more details, please refer to Appendix A.2. To train our MatchLM, we make use of Zero Redundancy Optimizer (ZeRO) Stage 2 [39] and LoRA [17] with a rank of 32, alpha 64 and dropout of 0.05.