Technical Summary:
Efficient Transfer From Image-Based Large Multimodal Models to Video Tasks

Kai-Yu Lu

2025/9/20

1 Research Problem and Motivation

This paper addresses the problem of efficiently extending image-based Large Multimodal Models (LMMs) to video understanding tasks. The core challenge is to exploit rich image-text pre-training while avoiding the substantial cost and potential performance degradation associated with conventional video pre-training stages.

Traditional video LMMs typically follow a three-stage pipeline: image pre-training, video pre-training, and task-specific video fine-tuning. Temporal modules are introduced and trained during the video pre-training phase. This design has several limitations for fine-grained, domain-specific downstream tasks, especially when data are scarce.

First, training temporal modules on large-scale video-caption corpora requires significant computational resources and is often unaffordable for many research groups. Second, adapting image-based encoders to video through heavy temporal pre-training may gradually overwrite fine-grained visual representations learned from diverse image-text data. This phenomenon reduces transfer performance when downstream tasks differ from pre-training objectives, for example video sarcasm detection or humor detection, where subtle facial expressions and dialog-level cues are crucial.

Third, there is a domain gap between conventional video captioning pre-training and many downstream tasks. Captioning emphasizes high-level event description, whereas intent analysis and egocentric question answering require more focused reasoning about emotions, intentions, and object-level temporal dynamics. Finally, the volume of video-text pre-training data is substantially smaller than that of image-text data, which can limit the generalization capability of video LMMs across scenes and domains.

The central motivation of the paper is therefore to design a temporal modeling and transfer framework that reuses image-based LMMs efficiently, preserves their strong visual-language knowledge, and achieves competitive or superior performance on small-scale, fine-grained video tasks without an explicit video pre-training stage.

2 Related Work

Research on multimodal video understanding for conversation and speech has focused on tasks such as emotion recognition, sarcasm detection, and humor detection. Earlier approaches typically combine Convolutional Neural Networks for visual features and Bidirectional Encoder Representations from Transformers (BERT) for text, followed by multimodal Transformers for fusion. These methods require attention modules to be trained from scratch and often suffer when training data are limited, because attention parameters are difficult to estimate reliably in low-data regimes.

With the advent of Large Language Models (LLMs), multimodal systems have increasingly used frozen LLM backbones combined with visual encoders. Representative image-based LMMs include BLIP-2, which uses a Q-former module to bridge frozen image encoders and frozen LLMs through image-text contrastive pre-training, and LLaVA, which maps visual tokens into the language representation space via lightweight adapters. These methods demonstrate that frozen LLMs can be effectively reused for multimodal reasoning when coupled with suitable visual alignment modules.

Video LLMs extend these ideas to video inputs. Models such as VideoLLaMA, VideoChat, and their successors introduce temporal modeling modules on top of visual encoders and LLM backbones, and they are typically pre-trained on large video-text corpora using video captioning or video question answering tasks. Although these models achieve impressive results, they incur high video pre-training cost and may not transfer well to tasks whose semantics differ significantly from captioning.

Temporal modeling in vision has also been explored outside the LLM context. Vision Transformers (ViTs), CLIP-based models, and other transformer architectures have been adapted to video by adding temporal attention layers or 3D convolutions. Parameter-efficient transfer techniques such as adapter-based tuning and Low-Rank Adaptation (LoRA) have been used to fine-tune a small subset of parameters while keeping the majority of the model frozen. Recent methods like AIM repurpose attention weights from image models for video action recognition, but they often neglect multimodal fusion and do not fully exploit pre-trained cross-modal attention.

The paper builds upon these lines of work by: (i) reusing pre-trained multi-modal Q-former attention weights for temporal modeling, (ii) performing early multimodal spatial-temporal fusion, and (iii) applying dynamic attention routing to adjust attention scopes, all under a parameter-efficient transfer framework that keeps the core LLM and most visual modules frozen. :contentReference[oaicite:0]index=0

3 Dataset Construction

The study does not create new datasets but instead reuses several established benchmarks in two categories: video intent analysis and out-of-pretraining-scene video question answering. For clarity, this section summarizes the main datasets, their modality configurations, and their roles in the experiments.

3.1 Video Intent Analysis Datasets

Video intent analysis focuses on recognizing speaker intent such as sarcasm or humor in dialogue-centered clips.

MUStARD

The Multimodal Sarcasm Detection Dataset (MUStARD) consists of short conversational video clips sourced from television shows such as “Friends” and “The Big Bang Theory”. Each sample includes a target utterance, its audio-visual segment, and preceding dialogue context. Labels indicate whether the target utterance is sarcastic. Two partition schemes are used: the original five-fold split and a train-development-test split, denoted as MUStARD*.

UR-Funnyv2

The UR-Funnyv2 dataset contains TED talk video segments annotated for humor. Each example includes the punchline utterance, its preceding context, and tri-modal observations (text, video, and audio). In addition to the original training split, the study defines a reduced training version to test performance under limited data. Labels indicate whether a segment is humorous.

3.2 Out-of-Pretraining Scene Datasets

To evaluate transferability to scenes that differ significantly from standard video pre-training corpora, three further datasets are used.

CLEVRER-MC

CLEVRER is a synthetic video dataset for physical reasoning involving collisions, motions, and object interactions. CLEVRER-MC is a multiple-choice variant that reformulates open-ended questions into discrete options, allowing objective evaluation. The paper utilizes tasks focused on object existence, motion direction, motion count, and motion attributes.

qaEgo4D

The qaEgo4D dataset is derived from the Ego4D egocentric video collection. Videos are captured from a first-person viewpoint and exhibit diverse environments and activities. Each clip is paired with a natural language question about recent events in the video, such as object locations or performed actions, together with a textual answer.

YouCook2

YouCook2 is a large instructional cooking video dataset with temporally segmented procedure steps described by imperative sentences. The study uses it in a question answering setting, where questions pertain to ingredients, actions, or upcoming steps within cooking procedures.

3.3 Dataset Summary Table

Table 1 summarizes the main characteristics of the datasets used in the experiments, including domain, modalities, approximate scale, task type, and label format.

Dataset	Domain	Modalities	Scale	Task / Labels
MUStARD	TV dialogue sarcasm	Video, text, audio	Small-scale	Binary sarcasm label
UR-Funnyv2	TED talk humor	Video, text, audio	Medium-scale	Binary humor label
CLEVRER-MC	Synthetic physics	Video, text	Medium-scale	Multiple-choice reasoning
qaEgo4D	Egocentric daily life	Video, text	Large-scale	Open-ended textual answers
YouCook2	Cooking procedures	Video, text	Large-scale	Open-ended textual answers

Table 1: Overview of datasets used for video intent analysis and out-of-pretraining-scene video question answering.

4 Query Protocol and Task Definitions

The study formulates all tasks as instruction-following problems for a frozen Large Language Model. A context sequence and a video clip are provided, and the model is trained to generate an answer in natural language. The context sequence contains task instructions, questions, and potentially dialogue history.

Let C denote the textual context and v denote the video clip. The model outputs an answer sequence C_o conditioned on both inputs. Different datasets instantiate this generic format in distinct ways.

For MUStARD and UR-Funnyv2, the task is video intent analysis. The context includes dialogue history and a natural language instruction such as “Classify whether the target utterance is sarcastic or not” or

"Determine whether this segment is humorous". The model must generate a discrete label, for example "sarcastic" or "not sarcastic", which can be mapped to binary classes for evaluation.

For CLEVRER-MC tasks, the context describes the question about the synthetic scene, for example "How many objects are moving in the video". The answer is restricted to one of several textual options corresponding to multiple-choice candidates. The evaluation is based on whether the generated answer matches the correct option.

For qaEgo4D and YouCook2, the protocol is open-ended video question answering. The context contains the question, possibly with additional instructions describing the expected answer format. The model outputs a free-form textual answer, which is scored using both exact correctness and text similarity metrics.

In all cases, training and evaluation rely on mapping natural language outputs to task-specific labels or similarity scores defined by the benchmark. The model treats all tasks through a unified generative interface, which also allows comparison with other LLM-based video understanding systems.

5 Modeling Approach

The proposed framework, named MTransLLAMA, transfers an image-based LMM to video tasks using three key components: multi-modal spatial-temporal early fusion, multi-modal query temporal reusing, and dynamic attention routing. The underlying image-based LMM is similar to BLIP-2, consisting of a vision encoder, a text encoder, a Q-former module for image-text fusion, and a frozen LLM.

5.1 Overall Formulation

Conventional video LMMs use a temporal module f_θ that receives visual embeddings and learnable queries, with text embeddings fed directly to the LLM. This interaction pattern is formalized as

$$C_o = \mathrm{LLM}\big(E_t, f_\theta(E_v, q)\big), \tag{1}$$

where C_o is the generated answer, E_t denotes text embeddings of the context, E_v denotes visual embeddings of the video, q denotes learnable multi-modal queries, and f_θ is a spatial-temporal module with trainable parameters θ. In this formulation text and visual features are fused relatively late in the pipeline.

The proposed MTransLLAMA changes this pattern by performing early multi-modal fusion inside the spatial-temporal module:

$$C_o = \mathrm{LLM}\big(E_t, f_\theta(E_t, E_v, q)\big). \tag{2}$$

In this equation the module f_θ receives both text embeddings E_t and visual embeddings E_v in addition to the query tokens q. This design allows text to guide visual feature extraction and temporal modeling from the earliest layers, which is particularly useful for intent analysis tasks that require focusing on semantically relevant regions.

5.2 Feature Extraction and Positional Encoding

The video clip v is represented as a tensor in $\mathbb{R}^{T\times 3\times H\times W}$, where T is the number of sampled frames, and each frame is an RGB image of height H and width W. Text embeddings E_t are produced by a BERT-style tokenizer and encoder.

Visual embeddings are computed frame by frame using a Vision Transformer (ViT):

$$E_v = \{x_i = \mathrm{ViT}(v_i) \mid i = 1, \ldots, T\}. \tag{3}$$

Here v_i is the i-th video frame, and x_i is the corresponding sequence of visual tokens for that frame. The number of visual tokens per frame is denoted by K_V, and each token has dimensionality D. Thus $E_v \in \mathbb{R}^{T \times K_V \times D}$.

The Q-former receives the visual embeddings E_v, the text embeddings E_t, and a set of learnable query tokens $q \in \mathbb{R}^{L \times D}$, where L is the number of queries. These queries are replicated across frames, yielding a tensor in $\mathbb{R}^{T \times L \times D}$. To encode spatial and temporal positions, the initial query representation is defined as

$$q_{i,l}^{(0)} = q_{i,l} + e_i^t + e_l^s, \tag{4}$$

where $q_{i,l}$ is the query embedding at frame index i and query index l, e_i^t is a learnable temporal positional embedding for frame i, and e_l^s is a learnable spatial positional embedding for query l. This equation injects information about both frame order and query index into the queries.

5.3 Multi-Modal Spatial-Temporal Early Fusion

Each layer of the Q-former is extended to perform three types of attention operations: spatial self-attention, cross-attention, and temporal self-attention. These operations implement early and recurrent spatial-temporal fusion.

Spatial self-attention jointly processes text embeddings and queries within a frame. In the m-th layer, spatial self-attention is written as

$$E_t', q' = \text{S-ATN}^L(E_t \cup q), \tag{5}$$

where S-ATN^L denotes self-attention applied along the token dimension that includes both text tokens and query tokens. The symbol $\cup$ denotes concatenation along the sequence axis. This operation allows textual information to influence the query representations and vice versa, achieving early text-visual fusion.

After spatial self-attention, cross-attention integrates visual embeddings into the query tokens:

$$q' = \text{C-ATN}^L(E_v, q), \tag{6}$$

where C-ATN^L is cross-attention along the query dimension. Here the queries attend to visual tokens from the corresponding frame, allowing the model to focus on spatial regions relevant to the joint textual and visual context.

Temporal self-attention is then applied across frames by reusing the same attention parameters, as described in the next subsection. By alternating spatial and temporal attention across layers, the model performs cyclic spatial-temporal fusion at multiple levels of abstraction.

5.4 Multi-Modal Query Temporal Reusing

A central innovation is the reuse of pre-trained image-text Q-former self-attention weights for temporal modeling. Instead of introducing a separate temporal module with new parameters, MTransLLAMA applies the same attention weights along the temporal dimension after a channel swapping operation.

Within the spatial attention module, query tensors are initially reshaped such that the batch size B and frame dimension T are merged, giving a tensor of shape $BT \times L \times D$. Self-attention and cross-attention are then computed along the query dimension L as in Equations (5) and (6).

For temporal modeling, the tensor is reshaped to group queries by identity rather than by frame. Concretely, the tensor is reshaped to $BL \times T \times D$, and self-attention is computed along the temporal dimension:

$$q' = \text{S-ATN}^T(q), \tag{7}$$

where S-ATN^T denotes self-attention applied across frames for each query identity. This operation captures temporal relationships of semantically similar tokens across time while reusing the same projection matrices as in spatial self-attention. The channel swapping strategy therefore converts a purely spatial attention mechanism into a temporal one without duplicating parameters.

To adapt the reused attention to video-specific patterns while preserving the original image-text knowledge, the method employs Low-Rank Adaptation (LoRA). LoRA introduces small trainable matrices that are added to the original attention weight matrices. For the query projection matrices in spatial and temporal attention, the adaptation is defined as

$$W_q^s = W_q + \mathrm{LoRA}_s(W_q), \quad W_q^t = W_q + \mathrm{LoRA}_t(W_q), \tag{8}$$

where W_q is the original query projection matrix from the pre-trained Q-former, W_q^s and W_q^t are the adapted matrices for spatial and temporal attention respectively, and $\mathrm{LoRA}_s(\cdot)$ and $\mathrm{LoRA}_t(\cdot)$ denote low-rank update functions parameterized by small trainable matrices. All original Q-former weights and the LLM remain frozen; only the LoRA parameters are trained. This design significantly reduces the number of trainable parameters while enabling effective temporal modeling.

5.5 Dynamic Attention Routing

Dynamic Attention Routing (DAR) is introduced to adjust the effective receptive field of attention. Different tasks require different attention spans: fine-grained intent detection may require localized attention on facial regions, while physical reasoning tasks may benefit from global attention across frames.

DAR defines a set of attention masks $\{M_0, \ldots, M_{p_k-1}\}$ with varying receptive fields for the k-th attention layer. These masks determine which token pairs are allowed to attend to each other. The layer then computes a weighted combination of attention matrices associated with each mask. For a given input q, the routed attention is formulated as

$$\mathrm{DAR}(q) = \big[W_q(E_t \cup q)\big]\big[W_k(E_t \cup q)\big]^\top \frac{1}{\sqrt{D_h}} \otimes \sum_{j=0}^{p_k-1} \alpha_j M_j, \tag{9}$$

where W_q and W_k are query and key projection matrices, D_h is the head dimension, and $\otimes$ denotes the element-wise (Hadamard) product. The masks M_j enforce different local or global attention patterns, and the routing probabilities α_j select and combine them.

The routing probabilities are obtained by applying a small network to a pooled summary of the query tokens:

$$\alpha = \mathrm{MLP}\big(\mathrm{APool}(q)\big), \tag{10}$$

where $\mathrm{APool}(\cdot)$ is a one-dimensional adaptive average pooling operator over the token dimension, and MLP is a two-layer feed-forward network with output dimensionality equal to the number of masks. This formulation allows the model to choose appropriate attention scopes based on the input content.

DAR is applied to both spatial and temporal attention modules, enabling adaptive control over spatial and temporal receptive fields across layers and samples.

5.6 Training Objective

During training, the model aims to generate an output answer C_o that matches the ground truth answer C_a at the token level. The textual answer C_a is first converted into a sequence of target tokens:

$$G = \mathrm{Tokenizer}(C_a), \tag{11}$$

where the tokenizer maps the natural language answer into the discrete vocabulary of the LLM. Let p_i denote the predicted probability distribution over the vocabulary at position i, and let G_i denote the one-hot target vector at the same position.

The training loss is the standard cross-entropy loss summed over token positions:

$$\mathcal{L} = -\sum_{i} G_i \log(p_i). \tag{12}$$

This loss encourages the model to assign high probability to the correct token at each position, which corresponds to minimizing the discrepancy between predicted and ground truth answer sequences. In intuitive terms, the loss measures how close the generated answer is to the reference answer, similar to grading each word in a student response against a standard solution.

6 Empirical Results

6.1 Experimental Setup

All experiments are conducted with a frozen Vicuna-7B language model and a Q-former pre-trained on image-text data following BLIP-2. Vision and text feature extractors are also frozen: a ViT acts as the visual encoder and a BERT-style encoder provides text embeddings. LoRA modules are inserted into the query and value projections of each Q-former attention head, and only these LoRA parameters are trained.

Training uses the AdamW optimizer with a learning rate of 0.0001, momentum parameters $(0.9, 0.95)$, and weight decay 0.05. For UR-Funnyv2, qaEgo4D, CLEVRER-MC, and YouCook2, the batch size is 2, the temporal window size is 4 frames, and the number of epochs is 25. For MUStARD and MUStARD*, a batch size of 4, window size of 8, and 90 epochs are used. All models are implemented in PyTorch and trained on Nvidia A40 GPUs with 48 GB memory.

6.2 Video Intent Analysis Results

On MUStARD and MUStARD*, performance is evaluated using binary accuracy, because the test sets are balanced between sarcastic and non-sarcastic classes. On UR-Funnyv2, accuracy is also used to measure humor detection performance.

The proposed MTransLLAMA clearly outperforms unimodal baselines. On MUStARD, it improves accuracy by approximately 19.6 percentage points over visual-only baselines and by approximately 7.7 percentage points over text-only baselines. On UR-Funnyv2, it achieves an improvement of about 9.8 percentage points over unimodal methods. These results confirm that combining video and text under the proposed architecture is highly beneficial for intent analysis.

Compared with strong multimodal baselines that use all three modalities (text, video, and audio), such as recent models specifically designed for sarcasm and humor detection, MTransLLAMA focuses only on text and video and freezes the audio branch. Even under this constraint, it improves intent analysis accuracy by around 2 percentage points on UR-Funnyv2, MUStARD, and MUStARD*. The results indicate that the proposed spatial-temporal fusion and temporal reusing strategies are particularly effective for recognizing subtle speaker intent.

When compared to LLM-based video understanding systems like VideoLLaMA and VideoChat under comparable LoRA fine-tuning settings, MTransLLAMA achieves additional gains on MUStARD, around 1 to 2 percentage points, despite not using any video pre-training. This demonstrates that efficient temporal adaptation on top of an image-based LMM can match or surpass fully video-pre-trained models for fine-grained intent tasks.

6.3 Out-of-Pretraining Scene Results

For CLEVRER-MC, the model is fine-tuned on the training split and evaluated on four tasks in MVBench: Object Existence (OE), Moving Direction (MD), Moving Count (MC), and Moving Attribute (MA). All tasks are treated as multiple-choice question answering. MTransLLAMA uses only about 1 percent of the available training videos and 0.5 percent of the total parameters as trainable LoRA weights.

Despite the extremely small trainable parameter budget and the lack of video pre-training, MTransLLAMA achieves accuracy comparable to VideoLLaMA2 on MA and OE tasks and performs similarly or better than VideoChat2 and the original VideoLLaMA on MD and MC. InternVL2 and other fully video-pre-trained LLMs still obtain the best scores on CLEVRER-MC, which is expected because the dataset is closer in nature to typical video pre-training corpora. Nevertheless, the results show that the proposed transfer method can reach competitive performance in synthetic physical reasoning tasks with much lower computational cost.

On qaEgo4D and YouCook2, the evaluation uses accuracy together with ROUGE and METEOR scores to account for text similarity between generated and reference answers. MTransLLAMA achieves state-of-the-art performance among LLM-based methods on these datasets, ranking just below the strongest video LMM InternVL2. Traditional non-generative baselines such as HCRN achieve higher accuracy in qaEgo4D by using predefined answer dictionaries and more frames, but they lack the flexibility of open-ended generation and are not based on large language models.

6.4 Efficiency and Parameter Analysis

Table 2 summarizes the cost profile of MTransLLAMA compared with several video LMMs. The main efficiency advantages arise from two factors. First, the method completely removes the video pre-training stage, which in other systems requires large video-text datasets such as WebVid-2M and consumes significantly more computational resources than fine-tuning. Second, by reusing Q-former parameters and training only LoRA modules, the number of trainable parameters is minimized.

Model	Video Pre-training	Trainable Params	Temporal Module Design
VideoLLaMA	Required	Large	Dedicated temporal Q-former
VideoChat2	Required	Large	Dedicated temporal modules
InternVL2	Required	Large	Joint video-text pre-training
MTransLLAMA	Not required	Very small (LoRA only)	Reused Q-former with channel swapping

Table 2: Qualitative comparison of efficiency between MTransLLAMA and representative video LMMs.

As the number of video frames increases, MTransLLAMA shows lower memory usage and algorithmic complexity compared with systems that add heavy temporal modules. Freezing both the projection layers and the LLM further simplifies fine-tuning and deployment, making the framework attractive for practitioners with limited hardware.

6.5 Ablation Study

Ablation experiments are conducted to assess the contributions of Multi-Modal Query Temporal Reusing (MQT), early fusion (UniFusion), and Dynamic Attention Routing (DAR). All ablations are implemented with the same Q-former and LLaMA-7B backbone.

Removing MQT and relying only on spatial feature extraction with concatenated image frames leads to a substantial performance drop on MUStARD and qaEgo4D, despite the use of more visual tokens and higher inference cost. This indicates that explicit temporal modeling through channel swapping and LoRA adaptation is critical for capturing dynamic cues.

The importance of early fusion is evaluated by comparing a dual-modal configuration with and without UniFusion. When text is introduced only after visual feature extraction, performance on UR-Funnyv2 and qaEgo4D is reduced. Early fusion, where text tokens enter the Q-former and participate in attention with visual tokens, yields superior results and reduces the number of tokens processed by the LLM, which improves efficiency.

Finally, disabling DAR and using a single fixed attention mask across layers results in consistent but moderate performance degradation across datasets. This confirms that allowing the model to route attention over multiple receptive fields based on the input offers tangible benefits, particularly for tasks that require simultaneous consideration of local details and global temporal context.

7 Summary

This technical summary has described an efficient framework, MTransLLAMA, for transferring image-based Large Multimodal Models to video understanding tasks under limited data and computational resources. The key idea is to reuse pre-trained multi-modal Q-former attention weights for temporal modeling, combined with early multi-modal spatial-temporal fusion and dynamic attention routing, while keeping the core LLM and most visual modules frozen.

The study shows that MTransLLAMA achieves strong performance on video intent analysis tasks such as sarcasm and humor detection, outperforming both unimodal baselines and several sophisticated multimodal systems, even when audio information is not fine-tuned. It also achieves competitive results on synthetic physical reasoning tasks and state-of-the-art or near state-of-the-art performance on egocentric and instructional video question answering among LLM-based models, all without video pre-training.

Limitations and Future Directions

A notable limitation is that when downstream tasks closely resemble the distributions of video pre-training datasets, such as CLEVRER-MC, fully video-pre-trained LMMs such as InternVL2 still outperform MTransLLAMA. This suggests that parameter-efficient temporal reuse may not fully substitute for dedicated video pre-training when abundant training data and computational resources are available.

Future research directions include combining the proposed transfer framework with lightweight video pre-training to further reduce the gap in highly structured tasks, extending the approach to include audio and additional modalities while retaining parameter efficiency, and investigating alternative routing mechanisms or attention sparsity patterns that further reduce computation without sacrificing fine-grained temporal reasoning.