Query-Guided Temporal Segment Refinement for Enhanced Multimodal Understanding

Xingjian Diao*, **Chunhui Zhang***, **Weiyi Wu, Zhongyu Ouyang, Peijun Qing, Ming Cheng, Soroush Vosoughi, Jiang Gui**
Dartmouth College
{xingjian.diao, chunhui.zhang, weiyi.wu}.gr@dartmouth.edu
{soroush.vosoughi, jiang.gui}@dartmouth.edu

Abstract

Multimodal foundation models (MFMs) have demonstrated significant success in tasks such as visual captioning, question answering, and image-text retrieval. However, these models face inherent limitations due to their finite internal capacity, which restricts their ability to process extended temporal sequences—an essential requirement for comprehensive video and audio analysis. To overcome these challenges, we introduce a specialized cognitive module, temporal working memory (TWM), which aims to enhance the temporal modeling capabilities of MFMs. It selectively retains task-relevant information across temporal dimensions, ensuring that critical details are preserved throughout the processing of video and audio content. The TWM uses a query-guided attention approach to focus on the most informative multimodal segments within temporal sequences. By retaining only the most relevant content, TWM optimizes the use of the model's limited capacity, enhancing its temporal modeling ability. This plug-and-play module can be easily integrated into existing MFMs. With our TWM, nine state-of-the-art models exhibit significant performance improvements across tasks such as video captioning, question answering, and video-text retrieval. By enhancing temporal modeling, TWM extends the capability of MFMs to handle complex, time-sensitive data effectively. Our code is available at https://anonymous.4open.science/r/ARRsubmission5.

1 Introduction

Multimodal foundation models (MFMs) have demonstrated impressive capabilities in a number of multimodal tasks, including visual caption (Kim et al., 2024; He et al., 2024), question-answer(Han et al., 2024), and image-text retrieval (Lin et al., 2024). Despite these advances, MFMs struggle to process multimodal inputs with extended temporal dimensions. This limitation can be ascribed to the working memory constraints in both human cognition (Baddeley, 2000) and foundation models (Zhang et al., 2024; Gong et al., 2024).

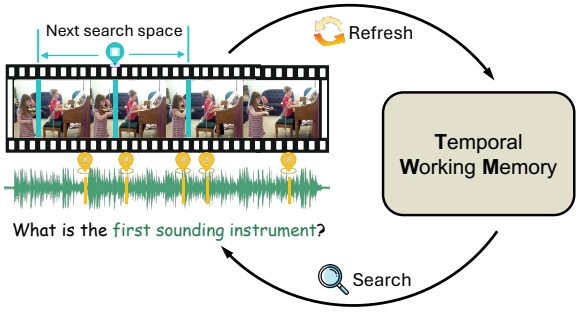

Figure 1: Temporal Working Memory (TWM): TWM employs search engines and memory refresh mechanisms to retain key segments in long multimodal inputs.

Multimodal foundation models (MFMs) have demonstrated impressive capabilities in tasks such as visual captioning, question-answering, and image-text retrieval. However, despite their promising capacity to process multimodal inputs, MFMs are often not equipped to explicitly reduce the input context burden, particularly in extracting query-relevant information from the input context. Unlike humans with working memory, MFMs lack mechanisms to selectively filter and retain only the most relevant temporal segments from multimodal inputs in the perceptual window (e.g., video frames or audio clips). Instead, current MFMs tend to process the entire input indiscriminately, leading to inefficient utilization of the model's capabilities. Despite these constraints, the working memory module employs efficient mechanisms, such as selective retention and distraction filtering, to prioritize relevant information while discarding irrelevant details. Inspired by these mechanisms, our proposed module seeks to address the limitations of MFMs by selectively retaining task-relevant tem-

*First two authors contributed equally.

poral segments (e.g., video frames or audio clips), thereby enabling the model to process extended temporal inputs effectively without overwhelming its computational resources.

In humans, working memory retains and processes information over short time spans with limited capacity, and similar constraints apply to MFMs (Liu et al., 2024a). For example, LLaMA has a context length limit of 2048 tokens (Touvron et al., 2023a,b; Dubey et al., 2024), while LLaVA (Liu et al., 2023) and BLIP-2 (Li et al., 2023b) can handle only 256 and 32 tokens per image, respectively. These limited capacities prevent models from effectively retaining sufficient information over extended temporal spans, such as those required for video comprehension and audio analysis. As memory saturation occurs, MFMs lose track of essential temporal details, significantly affecting their abilities on long-term context retention.

Recent developments in the memory of LLMs have significantly improved their ability to manage temporal contexts. Various approaches have been proposed to address the inherent memory limitations of LLMs and thereby improve their performance on complex tasks. Li et al. (2023a) proposed a knowledge-aware fine-tuning method that instructs LLMs to prioritize relevant external context while minimizing reliance on internal pretrained knowledge, thereby enhancing their memory by filtering out irrelevant information. Building on this, Gong et al. (2024) demonstrated that ChatGPT's working memory is similar to that of humans, suggesting that enhancing memory capacity is crucial for advancing the intelligence of AI systems. Further exploring these limitations, Zhang et al. (2024) recommended strategies for more efficient memory utilization, highlighting the importance of improving both memory retention and model autonomy for better reasoning capabilities. In the context of multimodal models, Wu and Xie (2024) integrated visual working memory mechanisms to help models focus on essential features in high-resolution images, significantly improving visual grounding performance.

Current approaches to working memory in LLMs, while effective for single-modal language tasks, struggle with dynamic multimodal and temporal reasoning: *(i)* these models are designed to handle only a single type of input, which limits their ability to combine and understand multiple types of data (e.g., audio, video, language). This limitation hinders their ability to achieve a comprehensive understanding in multimodal environments, where integrating different sensory inputs is essential. *(ii)* models lack the ability to effectively capture temporal dependencies. They struggle with short-term changes (e.g. rapid changes in audio or visual content) and long-term relationships (e.g. understanding how earlier events relate to later ones). This reduces their ability to reason about sequential and time-sensitive data, which is crucial for tasks involving events that unfold over time. *(iii)* these models struggle to efficiently extract relevant information from large amounts of raw data. This inefficiency leads to information overload, which strains the model's limited capacity and ultimately degrades performance, especially when dealing with complex and high-volume data.

Therefore, we draw on human working memory to effectively **extract query-relevant multimodal information across temporal dimensions**. We propose a Temporal Multimodal Memory (TMM) mechanism for MFMs, as shown in Figure 1. This mechanism employs a query-guided attention mechanism to selectively retain only query-relevant audio and visual inputs, focusing on the most informative segments along the temporal axis. The TMM mechanism constructs a temporal memory buffer at the model input stage, enabling the MFMs to efficiently store and manage critical information across time. By concentrating on the retention of the most relevant data, TMM significantly improves the model's ability to reason over extended temporal sequences in a multimodal context. Our contributions are:

- We propose a Temporal Working Memory (TWM) mechanism with an integrated query-guided selection module. This module directs the model to retain key segments in long video and audio sequences, optimizing the use of the model's limited capacity.
- We design a multi-scale temporal attention mechanism for both local and global dependencies, enabling accurate identification of relevant video-audio segments across temporal inputs.
- We integrate TWM into nine state-of-the-art MFMs and evaluate our approach on three large-scale multimodal benchmarks, covering tasks including audio-visual question answering, video captioning, and video-text retrieval. Our approach effectively yields significant performance improvements across all tasks.

2 Related Works

Temporal Modeling in MLLMs Multimodal LLMs (MLLMs) for long-video understanding aim to capture long-range temporal patterns. A common strategy is temporal pooling, as used in VideoChatGPT (Maaz et al., 2024), but this can limit performance due to inadequate temporal modeling. More advanced methods, such as video-LLAMA (Zhang et al., 2023), incorporate video query transformers to enhance temporal dynamics, but this comes with increased model complexity. To reduce computational demands, some models rely on pre-extracted features, avoiding joint training of backbone architectures (Hussein et al., 2019; Wu and Krahenbuhl, 2021). Techniques like Vis4mer (Islam and Bertasius, 2022) and S5 (Wang et al., 2023) utilize the S4 transformer architecture (Gu et al., 2022) for efficient long-range temporal modeling. Recent developments, such as online video processing (He et al., 2024), employ memory banks to track past content for long-term analysis. In contrast, we propose a TWM mechanism that retains only query-relevant multimodal inputs through search engines within a temporal context.

Video Understanding Video understanding tasks evaluate a model's ability to process multimodal content, focusing on both temporal and semantic aspects. Key tasks for long-video understanding include audio-visual question answering (AVQA), video captioning, and video-text retrieval, supported by extensive research and large-scale datasets (Liu et al., 2024b; Xu et al., 2016; Bain et al., 2020). Early AVQA methods fine-tuned pretrained visual models with adapters (Liu et al., 2024b; Lin et al., 2023; Duan et al., 2023), while recent approaches use unified multimodal encoders with LLMs (Han et al., 2024). Video captioning models employ graph neural networks (GNNs) (Hendria et al., 2023), simplified image-text architectures (Wang et al., 2022), or causal effect networks (CEN) to enhance temporal coherence (Nadeem et al., 2024). In video-text retrieval, adaptive frame aggregation reduces visual tokens to accelerate encoding (Ren et al., 2023). In contrast to previous work focusing on specific multimodal applications, this study emphasizes the role of TWM in enhancing fundamental temporal grounding across audio, video, and language.

Algorithm 1 TWM for MFMs

```
# search query-related segments from video&audio
def neural_search(video, audio, qry):
    # Step 1: Segment and encode video and audio
    v_seg, a_seg = segment(video, audio)
    v_embs, a_embs = encode(v_seg, a_seg)

    # Step 2: Calculate relevance scores
    v_scores, a_scores = sim(v_embs, a_embs, qry)

    # Step 3: Iterate to select the relevant
        segments
    v_buffer, a_buffer= select(v_embs, a_embs,
        v_scores, a_scores)

    return v_buffer, a_buffer

# temporal working memory for video and audio
v_buffer, a_buffer = neural_search(video, audio,
     qry)

# MFM with temporal working memory
output = MFM(v_buffer, a_buffer, qry)
```

3 Temporal Working Memory

Our temporal working memory (TWM) framework is a complementary architecture for Multimodal large language models, integrating visual and auditory buffer components. If the model does not involve audio, the auditory component can be omitted. The working pipeline of the TWM framework is outlined in Algorithm 1, while the search and memory update processes are depicted in Figure 2.

3.1 Visual Memory Management

3.1.1 Query-Relevant Video-Frame Search

Our mechanism emulates human cognitive strategies by identifying and retaining critical visual information from long video sequences. It alternates between two key operations—**search** and **update**—to dynamically adjust memory and focus on the most relevant segments, as shown in Figure 2.

Initially, k frames are selected from the full sequence of N frames and processed through a visual encoder to generate embeddings. Each frame v_i is assigned a **Similarity Score** ($S(v_i)$), defined as:

$$S(v_i) = D(v_i) + R(v_i, q), \tag{1}$$

where $D(v_i)$ is function representing the **distinctiveness** of frame v_i, $R(v_i, q)$ is function representing the **relevance** of frame v_i to the query q. The selection process is iterative. In each iteration, the frame with the highest similarity score ($S(v_i)$) is chosen as the midpoint, and k frames are searched uniformly within a range of $\frac{N}{k}$ around it. These frames are added to the visual memory, excluding

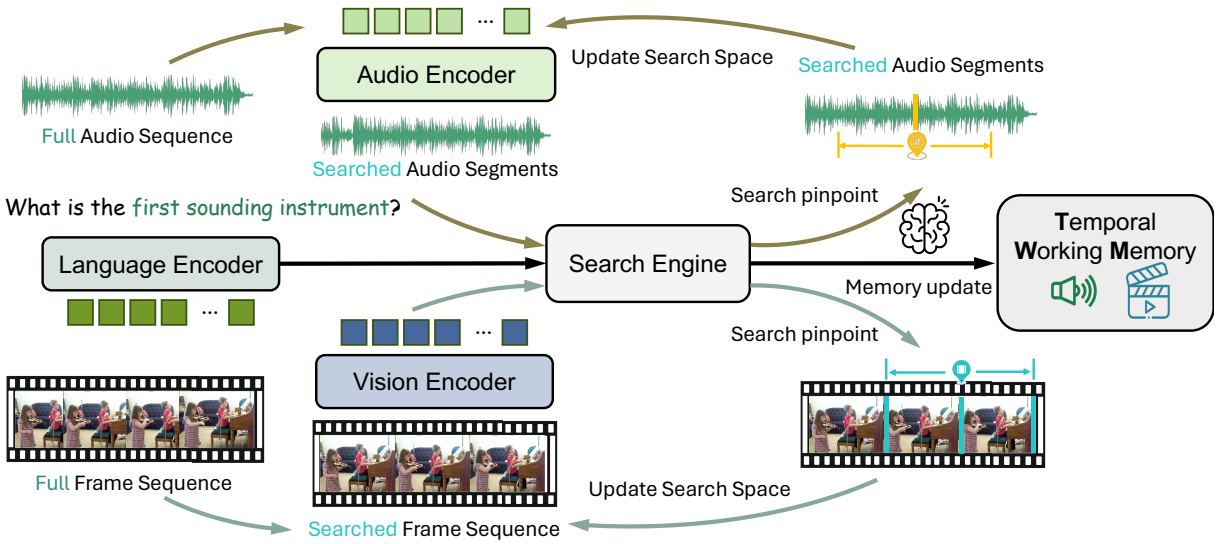

Figure 2: The temporal working memory (TWM) pipeline retains the most relevant segments from video and audio inputs based on a language query. The Language Encoder processes the query, guiding the Search Engine to identify and select key video and audio segments. TWM ensures the retention of only the most informative data, enabling the efficient utilization of multimodal foundation models' capabilities.

frames already present to maintain diversity. The process concludes upon convergence.

3.1.2 Training Neural Search Engine

To identify frames relevant to a given query, we use a **cross-modal alignment strategy** (Figure 3). Pre-trained visual and language encoders are employed, with a linear projection layer that maps visual embeddings into the textual embedding space. The **InfoNCE loss** (Oord et al., 2018) is used to optimize this alignment:

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp\left(\frac{\text{sim}(\mathbf{e}_v, \mathbf{e}_{t_i})}{\tau}\right)}{\sum_{j=1}^{N} \exp\left(\frac{\text{sim}(\mathbf{e}_v, \mathbf{e}_{t_j})}{\tau}\right)}, \quad (2)$$

where $\mathbf{e}_v$ is the embedding of video frame v, $\mathbf{e}_{t_i}$ is the embedding of text description t_i, $\text{sim}(x, y)$ denotes the cosine similarity function, τ is the temperature parameter controlling the sharpness of the distribution, and N represents the number of negative samples. The InfoNCE loss maximizes the similarity between corresponding video and text embeddings while minimizing similarity to unrelated samples. This ensures that the model effectively aligns the most relevant frames with their corresponding text, thereby optimizing its ability to retain meaningful visual information.

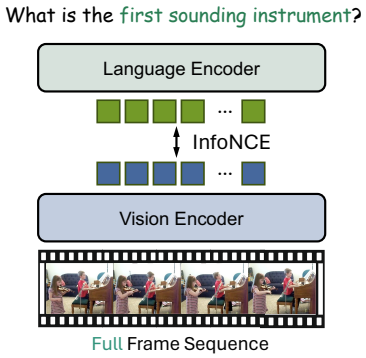

Figure 3: Aligning frames with language query. A linear projection layer trained with InfoNCE loss aligns visual embeddings with query-based anchors.

3.2 Auditory Memory Management

3.2.1 Query-Relevant Audio-Segment Search

The search process for identifying key audio segments mirrors the methodology used for video frames. The audio sequence is divided into predefined segments, typically 5-6 segments depending on video length, to model adequate temporal dependencies for tasks like AVQA and video captioning. This segmentation allows the model to focus efficiently on relevant audio intervals, improving attention allocation across extended sequences.

Building upon Pathformer's dual-attention mechanism (Chen et al., 2024), we extend their approach to enhance the correlation between audio and video data. Specifically, visual embeddings are used as queries in both attentions (Figure 4). To enable

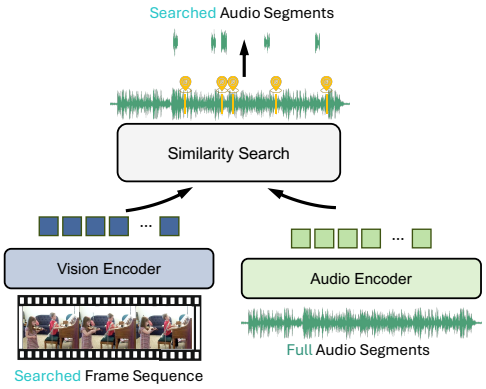

Figure 4: Similarity search for query-relevant audio segments. The audio encoder utilizes visual embeddings as a query to search for the most relevant audio segments, updating the auditory buffer to retain only the essential audio information.

multimodal audio-visual synchronization, we integrate audio patches derived from Mel-spectrograms for temporal segmentation. This facilitates alignment between audio and visual inputs. Our audio encoder employs two key attention mechanisms:

- Inter-segment attention is designed to model *global dependencies* across audio segments, enabling the model to capture broader relationships such as shifts in tone, mood, or overall sound context. Specifically, inter-segment attention calculates attention scores between the query Q, which is derived from visual features, and the keys K_i from the audio segments $Att_{\text{inter}} = \text{softmax}\left(\frac{QK_i^T}{\sqrt{d_K}}\right)V_i, \quad i \in [1, n]$. Here, Q represents the visual embeddings, while K_i and V_i are the audio embeddings from segment i. By using visual features as the query, this attention aligns audio information with relevant visual cues, effectively capturing how the audio context evolves to the video over time.

- Intra-segment attention aims to capture *local dependencies* within individual audio segments, thereby modeling fine-grained temporal patterns such as audio variations or sudden changes in sound effects. The result of intra-segment attention for each segment is computed as: $Att_{\text{intra}} = \text{Concat}(Att_{\text{intra}_i} \mid i = 1, \ldots, n)$. Att_{intra_i} represents the computed attention within each segment i. The concatenation operation aggregates these intra-segment attention results across all segments, ensuring that short-term changes are captured and preserved for subsequent processing. This aggregation allows the model to retain a detailed representation of the short-term dynamics within each segment.

In the fusion layer, we apply a cross-modal attention mechanism to synchronize features from both audio and visual streams. Additionally, multi-kernel pooling aggregates audio patches across different-scale temporal dependencies, enhancing the alignment and understanding of temporal multimodal inputs.

As shown in Figure 5, a pretrained visual feature extractor is used to align audio segments with their corresponding visual frames to establish cross-modal coherence. To identify query-relevant audio patches, we apply cosine similarity between the audio embeddings ($\mathbf{e}_{a_i}$) and visual embeddings ($\mathbf{e}_v$) as $\text{sim}(\mathbf{e}_{a_i}, \mathbf{e}_v) = \frac{\mathbf{e}_{a_i} \cdot \mathbf{e}_v}{\|\mathbf{e}_{a_i}\| \|\mathbf{e}_v\|}$. This similarity score is used to select the audio patches that are most relevant to the corresponding visual frames. The selected audio segments are then updated in the auditory buffer through an iterative process, ensuring that the most important audio information is retained. This iterative refinement enhances the synchronization and complementarity between audio and visual content.

3.2.2 Training Audio Search Engine

To identify query-relevant audio segments, we also use InfoNCE loss to achieve cross-modal alignment (see Figure 5). Let $\mathbf{e}_{a_i}$ denote the embedding of the audio patch a_i, and let $\mathbf{e}_v$ denote the embedding of the video frame v. The alignment loss is:

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp\left(\frac{\text{sim}(\mathbf{e}_v, \mathbf{e}_{a_i})}{\tau}\right)}{\sum_{j=1}^{N} \exp\left(\frac{\text{sim}(\mathbf{e}_v, \mathbf{e}_{a_j})}{\tau}\right)}, \quad (3)$$

where τ is a temperature parameter controlling the distribution's sharpness, and N is the number of negative samples. This approach ensures effective alignment between audio and visual embeddings, allowing the model to identify cross-modal relationships effectively and refine the auditory buffer.

4 Experiments

We perform three major experiments to validate the effectiveness of the Temporal Working Memory mechanism discussed in the previous section. We evaluated the performance of state-of-the-art baseline models and the same models augmented with our Temporal Working Memory mechanism on the following downstream tasks: (1) **audio-visual**

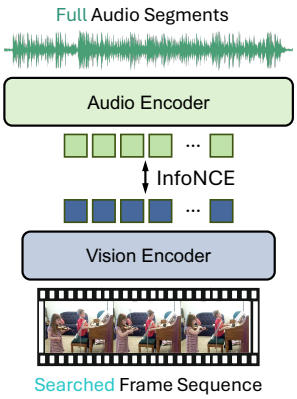

Figure 5: Audio segments aligned with query-relevant frames. An audio encoder learns temporal distance and resolution, aligning the visual-audio embeddings.

question answering (AVQA), (2) **video captioning**, and (3) **video-text retrieval**.

4.1 Setup

Datasets We conduct experiments on AVQA, video captioning, and video-text retrieval:

- **MUSIC-AVQA v2.0** (Liu et al., 2024b): MUSIC-AVQA v2.0 introduces 1,230 additional videos and 8,100 new question-answer (QA) pairs to further mitigate data bias. It builds on the original MUSIC-AVQA dataset (Li et al., 2022), which contains 9,288 videos of 22 musical instruments (over 150 hours) with 45,867 question-answer (QA) pairs across 33 templates in 9 categories.
- **MSR-VTT** (Xu et al., 2016): Comprises 10,000 video clips (over 41 hours) from various online sources. Each video has 20 human-annotated captions, totaling 200,000 video-text pairs across diverse categories like music, sports, and gaming.
- **CMD** (Bain et al., 2020): The Condensed Movies Dataset includes over 33,000 clips from 3,600 movies, averaging 2 minutes each, with annotations such as captions, dialogues, and action labels, ideal for video-text retrieval tasks.

Baselines Details of the baseline models can be found in Appendix A. We evaluate nine state-of-the-art MFMs reproduced with open-source code and pretrained weights.

Evaluation Metrics Below standard metrics reflect the accuracy, quality, and retrieval capabilities:

- **Audio-Visual Question Answering**: Accuracy is measured for Audio (Counting, Comparative), Visual (Counting, Location), Audio-Visual (Existential, Counting, Location, Comparative, Temporal) question types, along with average accuracies for Audio, Visual, Audio-Visual, and overall.
- **Video Captioning**: Metrics include ROUGE-L, CIDEr, and SPICE, assessing overlap with ground truth, consensus with human annotations, and diversity of scene descriptions, respectively.
- **Video-Text Retrieval**: Metrics are Recall@1, Recall@5, and Recall@10, measuring retrieval performance within top 1, 5, and 10 predictions.

Implementations The settings of each dataset:

- **MUSIC-AVQA v2.0**: Videos are 60 seconds long, with questions targeting specific portions of the video. We set $k = 11$ and ran 6 iterations, using $\alpha_1 = 0.2$ and $\alpha_2 = 0.8$, resulting in frame sampling rates consistent with 1 frame per second (fps). Audio segments are extracted every 5 seconds, selecting the highest-scoring segment from a total of 12 segments.
- **MSR-VTT**: With a frame rate of 21 fps, we set $k = 3$ and ran 3 iterations, yielding 8–9 searched frames, with $\alpha_1 = 0.5$ and $\alpha_2 = 0.5$ for balanced frame selection.
- **CMD**: With a frame rate of 30 fps, we used $k = 5$ and ran 7 iterations, producing 30–35 frames in total, using $\alpha_1 = 0.6$ and $\alpha_2 = 0.4$ to prioritize frame diversity.

As depicted in Figure 3, the dimensions of the linear mapping layer, and similarly in Figure 5 for the fusion layer, correspond to the output embedding dimensions of the text encoder used by each model. The embedding dimensions typically span 768-D, 4096-D, or 16384-D, as detailed in Section 4.1 where the baseline models are referenced.

Training is conducted on the NVIDIA H100 80GB GPUs using PyTorch. The Adam optimizer with a learning rate of $1e^{-4}$ is used, and each model is trained for 10 epochs. On the MUSIC-AVQA dataset, each model requires an average training time of 65.2 hours. For the MSR-VTT dataset, the average training time per model is 14.6 hours, while for the CMD dataset, each model takes approximately 146.6 GPU-hours to train.

4.2 Overall Comparisons

4.2.1 Audio-Visual Question Answering

TWM captures fine-grained multimodal dependencies for comparative reasoning In AVQA (Table 1), TWM excels in identifying and preserving fine-grained dependencies between audio and visual inputs, especially in complex comparative tasks. In audio-related comparative QA, LAVisH+TWM improves by 12.40%, DG-SCT+TWM

Method	Audio-related QA			Visual-related QA			Audio&Visual-related QA						Avg
	Count	Comp	Avg	Count	Local	Avg	Exist	Count	Local	Comp	Temp	Avg	
LAVisH (Lin et al., 2023)	**83.82**	58.19	75.72	82.81	81.73	82.30	73.26	**73.45**	65.64	64.26	**60.82**	67.75	73.28
LAVisH + TWM	79.22	**70.59**	**76.91**	**84.52**	**84.10**	**84.31**	**78.66**	65.21	**73.05**	**74.34**	57.24	**74.10**	**74.42**
Gain (Δ)	-4.60	+12.40	+1.19	+1.71	+2.37	+2.01	+5.40	-8.24	+7.41	+10.08	-3.58	+6.35	+1.14
DG-SCT (Duan et al., 2023)	**83.13**	62.54	76.62	81.61	82.76	82.19	83.43	72.70	64.65	64.78	**67.34**	70.38	74.53
DG-SCT + TWM	80.05	**72.66**	**77.20**	**87.77**	**85.24**	**86.35**	**88.69**	**87.21**	**76.06**	**78.34**	59.82	**78.96**	**79.22**
Gain (Δ)	-3.08	+10.12	+0.58	+6.16	+2.48	+4.16	+5.26	+14.51	+11.41	+13.56	-7.52	+8.58	+4.69
LAST-Att (Liu et al., 2024b)	**86.03**	62.52	**79.44**	84.12	84.01	84.05	76.21	75.23	68.91	65.60	**60.60**	69.04	75.44
LAST-Att + TWM	79.52	**74.25**	76.43	**88.52**	**85.98**	**87.01**	**80.12**	**82.40**	**76.06**	**76.52**	55.82	**74.05**	**77.96**
Gain (Δ)	-6.51	+11.73	-3.01	+4.40	+1.97	+2.96	+3.91	+7.17	+7.15	+10.92	-4.78	+5.01	+2.52

Table 1: Results of different models on the test set of MUSIC-AVQA 2.0 (Liu et al., 2024b). **Bold** results indicate the better performance.

gains 10.12% and LAST-Att+TWM shows an increase of 11.73%. Similarly, significant improvements are observed in the audiovisual comparative QA: LAVisH+TWM improves by 10.08%, DG-SCT+TWM by 13.56% and LAST-Att+TWM by 10.92%. TWM also leads to significant increases in overall average accuracy across all audiovisual tasks, with LAVisH+TWM improving by 6.35%, DG-SCT+TWM by 8.58% and LAST-Att+TWM by 5.01%. By focusing on query-relevant segments and filtering out irrelevant content, TWM ensures that the model attends to the most informative parts of each modality, thereby improving cross-modal reasoning accuracy. This selective attention mechanism allows the model to better isolate critical elements within audiovisual streams, enabling it to reason about context-dependent relationships between different inputs. The fine-tuned multimodal focus highlights TWM's effectiveness in tasks requiring fine-grained comparisons, as it actively suppresses noise and enhances relevant signals.

4.2.2 Video Captioning

Method	**ROUGE-L ↑**	**CIDEr ↑**	**SPICE ↑**
Git (Wang et al., 2022)	24.51	32.43	13.70
Git + TWM	**26.10**	**39.25**	**14.31**
Gain (Δ)	+1.59	+6.82	+0.61
AKGNN (Hendria et al., 2023)	**21.42**	25.90	**11.99**
AKGNN + TWM	21.33	**27.46**	11.02
Gain (Δ)	−0.09	+1.56	−0.97
CEN (Nadeem et al., 2024)	27.90	49.87	15.76
CEN + TWM	**28.10**	**52.01**	**15.90**
Gain (Δ)	+0.20	+2.14	+0.14

Table 2: Test results of different models on MSR-VTT.

TWM enhances temporal coherence through selective attention to key events In video captioning tasks (Table 2), TWM's selective attention mechanism significantly improves temporal coherence by focusing on key events and discarding irrelevant content. For example, Git+TWM achieves a 6.82% improvement in CIDEr and a 1.59% increase in ROUGE-L, highlighting the model's enhanced ability to generate coherent narratives that follow the flow of events. By retaining only the most contextually relevant audio-visual segments, TWM helps to avoid disjointed or fragmented scene descriptions, which is critical for accurately representing long or complex narratives.

TWM captures fine-grained scene transitions, enhancing descriptive richness TWM also excels at capturing fine-grained details within scenes, allowing the model to generate richer and more informative descriptions. For example, CEN+TWM shows a 0.14% improvement in SPICE, reflecting the model's enhanced ability to capture varied and accurate content in captions. In addition, Git+TWM shows a 0.61% increase in SPICE, indicating an improved ability to capture changes in the audiovisual content of the video, such as changes in action or context. By dynamically updating memory with the most relevant visual and auditory elements, TWM ensures that critical details are highlighted, resulting in more contextually accurate and detailed output. This enhanced descriptive richness is essential in scenarios involving complex or rapidly changing scenes, where capturing semantic shifts in the narrative is key to generating informative captions.

4.2.3 Video-Text Retrieval

TWM maintains retrieval performance across broader scopes through adaptive segment retention TWM's ability to enhance cross-modal alignment extends beyond immediate retrieval precision to broader retrieval tasks (Table 3). For instance, VINDLU + TWM achieves improvements of 2.1% in Recall@1, 1.8% in Recall@5, and 2.8%

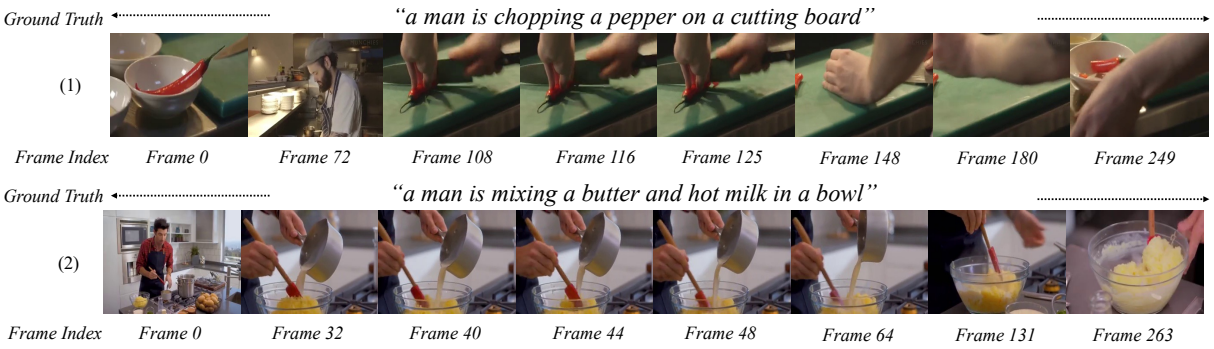

Figure 6: TWM-searched frames for the video captioning task, integrated with AKGNN (Hendria et al., 2023). Two examples of frames searched by TWM are presented alongside their ground truth captions. The total number of frames per clip in MSR-VTT (Xu et al., 2016) ranges from 210 to 630. TWM-searched 8 frames effectively encapsulate the key visual information required to generate the ground truth captions.

Method	Recall@1 ↑	Recall@5 ↑	Recall@10 ↑
VINDLU (Cheng et al., 2023)	18.4	36.4	41.8
VINDLU + TWM	**20.5**	**38.2**	**44.6**
Gain (Δ)	+2.1	+1.8	+2.8
TESTA (Ren et al., 2023)	21.5	42.4	50.7
TESTA + TWM	**22.1**	**45.3**	**52.1**
Gain (Δ)	+0.6	+2.9	+1.4
MovieSeq (Lin et al., 2024)	25.8	45.3	50.3
MovieSeq + TWM	**27.5**	**47.0**	**51.1**
Gain (Δ)	+1.7	+1.7	+0.8

Table 3: Test results of different models on CMD.

in Recall@10. TESTA + TWM demonstrates gains of 2.9% in Recall@5 and 1.4% in Recall@10, showcasing TWM's capacity to retain relevant segments even in complex or diverse video datasets. Similarly, MovieSeq + TWM shows consistent improvements with a 1.7% increase in Recall@1 and Recall@5, and a 0.8% gain in Recall@10. These results indicate that TWM's memory update mechanism is flexible enough to adapt to a wide range of retrieval tasks. By selectively focusing on the most important audio-visual elements, TWM improves the model's ability to retrieve relevant content across larger candidate sets. This adaptive retention mechanism allows the model to effectively balance precision and scope, ensuring that both specific and broad retrieval queries benefit from TWM's selective attention and memory update strategies.

4.3 The Impacts of Temporal Sequence Selection

To illustrate the effectiveness of TWM, we present a case study where TWM's search engine selects highly relevant frames based on the input of the language query (Figure 6). Below, we provide a detailed analysis of how TWM ensures completeness of action representation, eliminates irrelevant noise, and optimizes model performance through selective frame reduction.

Completeness of action representation TWM captures all the core stages of primary actions like chopping or mixing. For the chopping example, it captures the pepper being placed on the cutting board (Frames 0 and 72), the initiation of slicing (Frames 108 and 116), intermediate chopping stages (Frames 125 and 148), and the completion of the task (Frames 180 and 249). In the mixing example, it includes frames that depict the pouring of ingredients (Frames 40 and 44), the stirring process (Frames 48, 64, and 131), and the final state of the mixture. By covering all key moments, TWM provides the captioning model with a comprehensive understanding of the actions.

Elimination of irrelevant noise In cooking videos, various elements can distract from the main actions, such as the kitchen background, other utensils, or idle moments unrelated to the primary tasks like chopping. By selecting only the frames that focus on essential actions such as chopping or mixing in the examples, the TWM efficiently filters out these distractions, providing cleaner visual information for the captioning model. By minimizing distracting frames in the captioning inputs, the resulting descriptions of the events in the video become more accurate and useful.

Selective frame reduction for efficient model capacity utilization By selecting a limited number of informative frames that encompass all core stages, TWM optimizes memory usage and com-

putational resources. It directs computational resources to the most relevant parts of the sequence by eliminating redundant or unrelated frames, thereby enhancing performance without overburdening the model. This selective retention not only allows for the precise capture of essential actions, but also facilitates efficient processing, significantly saving memory and speeding up the captioning model. Additional case studies on video-text retrieval and audio-visual question answering tasks are available in the Appendix B.

5 Conclusion

We introduce Temporal Working Memory (TWM), a cognitive module designed to enhance multimodal foundation models by retaining only the most essential segments in complex video-audio-language tasks. Multimodal models often struggle with dynamic multimodal input due to limited internal capacity. TWM addresses these limitations by *(i)* expanding memory to process temporal multimodal sequences and identify key segments; *(ii)* modeling multi-scale temporal dependencies between video and audio inputs; and *(iii)* extracting query-relevant information from rich multimodal data for efficient memory utilization. Our approach effectively improves nine state-of-the-art multimodal models in three different temporal tasks.

Limitations

The effectiveness of TWM has been demonstrated on specific multimodal tasks and benchmarks, such as video captioning, question answering, and video-text retrieval. However, its generalizability to other in-the-wild domains remains unexplored. Extending TWM's application to other more practical multimodal tasks would be future direction for real-world applications.

Ethical Considerations

We examined the study describing the publicly available datasets used in this research and identified no ethical issues regarding the datasets.

References

Alan Baddeley. 2000. The episodic buffer: a new component of working memory? *Trends in cognitive sciences*.

Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. 2020. Condensed movies: Story based retrieval with contextual embeddings. In *Asian Conference on Computer Vision*.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. 2024. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *International Conference on Learning Representations*.

Feng Cheng, Xizi Wang, Jie Lei, David Crandall, Mohit Bansal, and Gedas Bertasius. 2023. Vindlu: A recipe for effective video-and-language pretraining. In *Conference on Computer Vision and Pattern Recognition*.

Haoyi Duan, Yan Xia, Zhou Mingze, Li Tang, Jieming Zhu, and Zhou Zhao. 2023. Cross-modal prompts: Adapting large pre-trained models for audio-visual downstream tasks. In *Advances in Neural Information Processing Systems*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dongyu Gong, Xingchen Wan, and Dingmin Wang. 2024. Working memory capacity of chatgpt: An empirical study. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Albert Gu, Karan Goel, and Christopher Ré. 2022. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*.

Jiaming Han, Kaixiong Gong, Yiyuan Zhang, Jiaqi Wang, Kaipeng Zhang, Dahua Lin, Yu Qiao, Peng Gao, and Xiangyu Yue. 2024. Onellm: One framework to align all modalities with language. In *Conference on Computer Vision and Pattern Recognition*.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. 2024. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *Conference on Computer Vision and Pattern Recognition*.

Willy Fitra Hendria, Vania Velda, Bahy Helmi Hartoyo Putra, Fikriansyah Adzaka, and Cheol Jeong. 2023. Action knowledge for video captioning with graph neural networks. *Journal of King Saud University-Computer and Information Sciences*.

Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. 2019. Timeception for complex action recognition. In *Conference on Computer Vision and Pattern Recognition*.

Md Mohaiminul Islam and Gedas Bertasius. 2022. Long movie clip classification with state-space video models. In *European Conference on Computer Vision*.

Byoungjip Kim, Dasol Hwang, Sungjun Cho, Youngsoo Jang, Honglak Lee, and Moontae Lee. 2024. Show think and tell: Thought-augmented fine-tuning of large language models for video captioning. In *Conference on Computer Vision and Pattern Recognition*.

Daliang Li, Ankit Singh Rawat, Manzil Zaheer, Xin Wang, Michal Lukasik, Andreas Veit, Felix Yu, and Sanjiv Kumar. 2023a. Large language models with controllable working memory. In *Findings of the Association for Computational Linguistics: ACL*.

Guangyao Li, Yake Wei, Yapeng Tian, Chenliang Xu, Ji-Rong Wen, and Di Hu. 2022. Learning to answer questions in dynamic audio-visual scenarios. In *Conference on Computer Vision and Pattern Recognition*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023b. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*.

Kevin Qinghong Lin, Pengchuan Zhang, Difei Gao, Xide Xia, Joya Chen, Ziteng Gao, Jinheng Xie, Xuhong Xiao, and Mike Zheng Shou. 2024. Learning video context as interleaved multimodal sequences. In *European Conference on Computer Vision*.

Yan-Bo Lin, Yi-Lin Sung, Jie Lei, Mohit Bansal, and Gedas Bertasius. 2023. Vision transformers are parameter-efficient audio-visual learners. In *Conference on Computer Vision and Pattern Recognition*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024a. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*.

Xiulong Liu, Zhikang Dong, and Peng Zhang. 2024b. Tackling data bias in music-avqa: Crafting a balanced dataset for unbiased question-answering. In *Winter Conference on Applications of Computer Vision*.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. 2024. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Annual Meeting of the Association for Computational Linguistics*.

Asmar Nadeem, Faegheh Sardari, Robert Dawes, Syed Sameed Husain, Adrian Hilton, and Armin Mustafa. 2024. Narrativebridge: Enhancing video captioning with causal-temporal narrative. *arXiv preprint arXiv:2406.06499*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Shuhuai Ren, Sishuo Chen, Shicheng Li, Xu Sun, and Lu Hou. 2023. TESTA: Temporal-spatial token aggregation for long-form video-language understanding. In *Findings of the Association for Computational Linguistics: EMNLP*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. 2022. Git: A generative image-to-text transformer for vision and language. *Transactions on Machine Learning Research*.

Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. 2023. Selective structured state-spaces for long-form video understanding. In *Conference on Computer Vision and Pattern Recognition*.

Chao-Yuan Wu and Philipp Krahenbuhl. 2021. Towards long-form video understanding. In *Conference on Computer Vision and Pattern Recognition*.

Penghao Wu and Saining Xie. 2024. V*: Guided visual search as a core mechanism in multimodal llms. In *Conference on Computer Vision and Pattern Recognition*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Conference on Computer Vision and Pattern Recognition*.

Chunhui Zhang, Yiren Jian, Zhongyu Ouyang, and Soroush Vosoughi. 2024. Scaling cognitive limits: Identifying working memory limits in llms. In *Conference on Empirical Methods in Natural Language Processing*.

Hang Zhang, Xin Li, and Lidong Bing. 2023. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Conference on Empirical Methods in Natural Language Processing*.

A Baselines

We evaluate nine MFMs reproduced with open-source code and pretrained weights:

- **LAVisH** (Lin et al., 2023): LAVisH adapter uses a small set of latent tokens, forming an attention bottleneck that reduces the quadratic cost

of standard cross-attention. The trained LAVisH module alongside Swin-v2 serves as TWM visual encoder.

- **DG-SCT** (Duan et al., 2023): Adds cross-modal interaction layers to pretrained audio-visual encoders for adaptive extraction across spatial, channel, and temporal dimensions. All DG-SCT and Swin-T blocks serve as TWM visual encoder.
- **LAST-Att** (Liu et al., 2024b): Explores the interrelationships between audio-visual-text modalities. Swin-v2 serves as TWM visual encoder.
- **Git** (Wang et al., 2022): Simplifies the architecture to a single image encoder and a text decoder under a unified language modeling task. The pretrained image encoder serves as TWM visual encoder.
- **AKGNN** (Hendria et al., 2023): Introduces a grid-based node representation, where nodes are represented by features extracted from a grid of video frames. The trained graph neural network from AKGNN serve as TWM visual encoder.
- **CEN** (Nadeem et al., 2024): Utilizes independent encoders to capture causal dynamics and generate time-sequenced captions. The pretrained CLIP-ViT from CEN serves as TWM visual encoder.
- **VINDLU** (Cheng et al., 2023): Develops a stepwise approach for efficient VidL pretraining. The trained video encoder from VINDLU serves as TWM visual encoder.
- **TESTA** (Ren et al., 2023): Compresses video semantics by adaptively aggregating similar frames and similar blocks within each frame. The full video encoder blocks from TESTA serve as TWM visual encoder.
- **MovieSeq** (Lin et al., 2024): Through instruction tuning, MovieSeq enables a language model to interact with videos using cross-modal instructions. CLIP vision encoder from MovieSeq serve as TWM visual encoder.

B Additional Case Studies

B.1 Case Study: TWM Efficiently Retrieves Semantically Relevant Content for Video-Text Retrieval

In this case study (Figure 7), we demonstrate how the Temporal Working Memory (TWM) mechanism efficiently selects critical frames to accurately align video content with a textual query, effectively handling long video sequences and eliminating irrelevant noise. Using an example from the MovieSeq dataset, we analyze how TWM's frame selection and context inference contribute to its effectiveness.

Semantic Accuracy The retrieved frames effectively capture the key elements of the query. Frames 95, 114, 131, 137, 153, and 158 predominantly depict scenes where numerous employees are gathered in an office environment, attentively listening to a man in a suit making introductions. This indicates that TWM successfully understood the semantic concept of *employees* in an office setting and extracted relevant frames accordingly. Furthermore, frames 59, 64, 71, 78, 95, 114, 410, 418, 428, 457, 465, 474, 565, 582, 617, 929, and 1360 show the man in the suit holding a beaver puppet while speaking, demonstrating that the model accurately captured information related to *The Beaver*. By focusing on frames that directly correspond to the key entities and actions in the query, TWM ensures that the most relevant semantic content is brought to the forefront.

Inference of Implicit Information Interestingly, frame 53 was retrieved despite the absence of explicit appearances of *employees* or *The Beaver*. This suggests that TWM recognized the man in the suit as *Walter*, highlighting the mechanism's ability to infer relevant context even when explicit cues are missing. By including frames like this, TWM enhances the coherence and continuity of the searched content, as it captures the transitions and connections between key events. This allows for a more fluid and contextually rich representation of the video, as opposed to a disjointed collection of isolated moments. TWM evaluates frames not only for their direct relevance but also for their contextual significance within the narrative, ensuring that the searched frames maintain a logical progression and effectively convey the story. This ability to capture implicit relationships and provide a more comprehensive understanding of the video content sets TWM apart from methods that rely solely on explicit visual cues.

Searching Efficiency Due to TWM's preference for distinctiveness, the searched frames are highly informative with low redundancy. The mechanism assesses each frame's uniqueness and relevance,

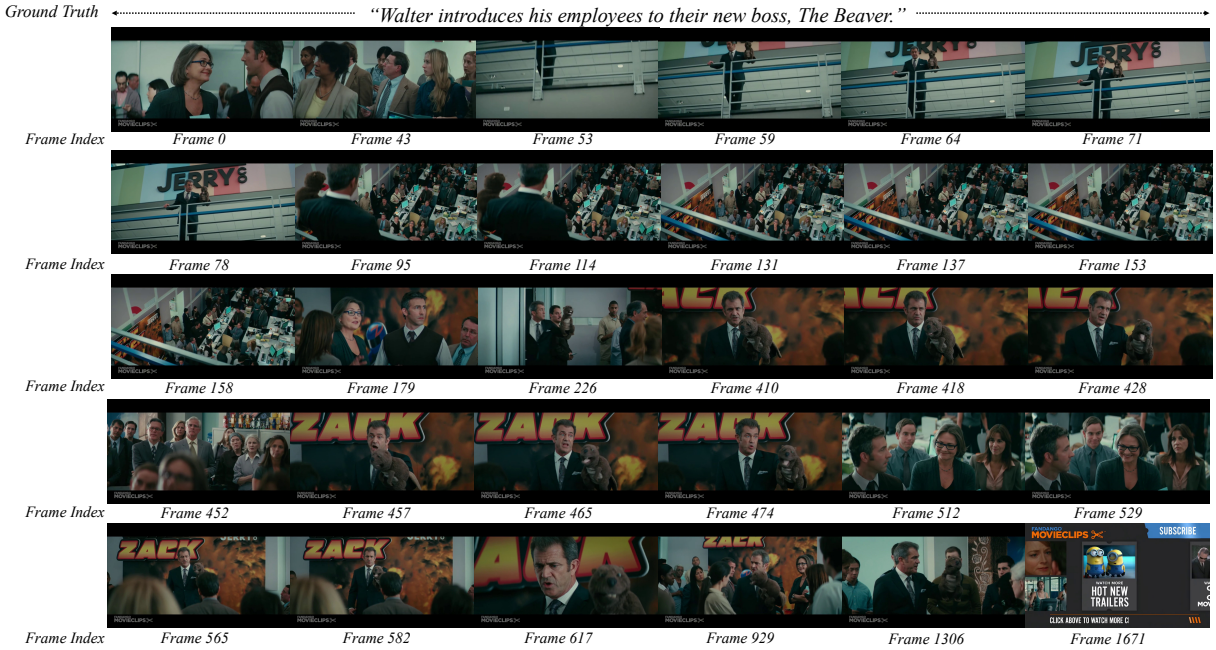

Figure 7: TWM-searched frames for the video-text retrieval task, integrated with Movieseq (Lin et al., 2024). This example showcases frames searched by TWM along with their corresponding ground truth text description. On average, each video clip in the CMD dataset (Bain et al., 2020) consists of 3,600 frames, and TWM has selected 30 key frames that effectively capture the relevant visual information.

prioritizing those that add new information rather than repetitive content. This efficient sampling enables the reconstruction of the narrative described in the query through a concise set of frames, enhancing computational efficiency and focusing the model's attention on critical moments. By reducing redundancy, TWM ensures that the model processes only the most significant frames, which improves both speed and accuracy in video-text retrieval tasks.

Effectiveness on Long Clips The searched frames demonstrate TWM's ability to handle long video sequences effectively. By maintaining relevance and diversity in frame selection across an extended temporal span, TWM provides comprehensive coverage of the pertinent content without being overwhelmed by the video's length. This scalability is crucial for real-world applications where videos often contain lengthy and complex narratives. TWM's mechanism allows it to sift through extensive footage and distill it into essential segments that align with the textual query.

Adaptive Relevance and Distinctiveness TWM achieves this effective retrieval through an iterative frame selection process that balances *relevance* to the query and *distinctiveness* within the video. By dynamically evaluating each frame based on its semantic content and its uniqueness in the sequence, TWM prioritizes frames that contribute most to understanding the video in the context of the query. This adaptive mechanism ensures that critical elements like *Walter*, *employees*, and *The Beaver* are included, while redundant or irrelevant frames are excluded. The ability to adjust selection criteria based on both the content and the query allows TWM to efficiently navigate and summarize long videos, enhancing the model's capacity for accurate and efficient video-text retrieval.

B.2 Case Study: TWM-Enhanced Frame Selection for Audio-Visual Question Answering (AVQA)

In this case study (Figure 8), we focus on the application of Temporal Working Memory (TWM) integrated with LAVisH (Lin et al., 2023) to solve an audio-visual question answering (AVQA) task using the MUSIC-AVQA v2.0 dataset (Liu et al., 2024b). The example task involves answering the question, *"Where is the loudest instrument?"* By employing TWM's search engine, the model selects and processes 60 frames from the video, filtering out irrelevant information and focusing on key segments with high audio-visual relevance.

Problem Context and TWM's Search Engine The AVQA task requires the model to identify the visual location of the instrument that generates the loudest sound. This type of task necessitates a tight integration between the visual and auditory modalities, as the model needs to accurately map auditory cues (i.e., loudness) to corresponding visual frames (i.e., instrument locations). TWM is specifically designed to address this multimodal challenge by efficiently searching through the sequence of video frames and retaining only the most task-relevant information in its memory. In this case, the memory focuses on frames where the auditory signal is strongest, i.e., where the loudest instrument plays, and where the visual context aligns with that information. The LAVisH module further enhances TWM's capabilities by providing latent token-based attention bottlenecks that reduce the computational cost of full cross-attention between the audio and visual streams. This makes it possible for TWM to focus its memory resources on the most relevant frames without overwhelming the model with unnecessary data.

Frame Selection Process The TWM's query-guided search engine is the key to efficiently selecting the 60 frames used for answering the query. The process begins by segmenting the video and aligning the visual and audio streams using a cross-modal attention mechanism. For this case, TWM identifies key frames, such as Frame 1593, Frame 1683, and Frame 1799, which correspond to moments where the audio peaks—indicating the presence of the loudest instrument. The frames selected by TWM exhibit both high auditory intensity and clear visual depictions of the instruments. This combination allows the model to not only identify when the sound is loudest but also to pinpoint which instrument is producing that sound and where it is located within the frame.

Noise Elimination and Memory Optimization One of TWM's primary strengths is its ability to eliminate irrelevant frames, ensuring that the model processes only those segments that are critical for answering the query. In the context of this AVQA task, many frames in the video may not contain significant sound events or may depict irrelevant background activities. TWM avoids these distractions by focusing on frames that feature high-intensity sound events and aligning them with visual cues of the instruments. For instance, TWM systematically ignores frames where the volume is low or where the instrument is not visually prominent. This significantly reduces the model's memory footprint, as only 60 frames from the full video sequence (which typically contains several hundred or even thousands of frames) are retained in memory. By focusing exclusively on the most relevant frames, TWM enhances both the efficiency and accuracy of the AVQA process.

Audio-Visual Alignment and Temporal Consistency TWM also excels in maintaining temporal consistency, ensuring that the selected frames accurately represent the progression of the event (i.e., playing the loudest instrument). The frames selected by TWM (e.g., Frame 1593, Frame 1683, Frame 1799) span the key moments of the audio event, capturing not just a single instant of loudness, but the broader temporal context surrounding the instrument's performance. This temporal aspect is crucial for accurately answering the question, as the model needs to understand the sequence of events—when the instrument starts playing, when it reaches its loudest point, and how the sound decays. TWM's memory update mechanism ensures that these temporal dynamics are preserved, allowing the model to reason over extended sequences without losing track of critical moments.

Localizing the Loudest Instrument In the context of this specific query—*"Where is the loudest instrument?"*—TWM's selection mechanism allows the model to accurately localize the instrument that produces the loudest sound. The key frames identified by TWM (Frames 1593, 1683, 1799) all show clear visual representations of the instrument in question, and the corresponding audio signals confirm that these are the moments of highest volume. By focusing on these high-relevance frames, the model can correctly answer the question, not only by identifying which instrument is the loudest but also by providing a precise visual reference for where it is located within the frame. This capability is critical for AVQA tasks, where both the audio and visual streams must be processed in tandem to generate a correct response.

Efficiency Gains and Computational Savings The integration of TWM into this AVQA task not only improves the model's accuracy but also provides significant efficiency gains. By reducing the number of frames processed from potentially hundreds or thousands down to just 60, TWM reduces the computational load on the model. This enables

faster inference times and more efficient use of resources, making it feasible to deploy AVQA models in real-time or resource-constrained environments. Moreover, TWM's memory optimization ensures that only the most relevant information is retained in memory, further reducing the computational cost associated with storing and processing large video datasets. This selective memory retention is a key advantage of TWM over traditional models that attempt to process the entire video sequence without filtering out irrelevant segments. Through this case study, we see how TWM enhances the performance of the model on a challenging AVQA task, making it more efficient, accurate, and capable of handling complex multimodal inputs.

Figure 8: TWM-searched frames for the audio-visual question answering task, integrated with LAVisH (Lin et al., 2023). This example illustrates 60 key frames searched by TWM from the MUSIC-AVQA v2.0 dataset (Liu et al., 2024b), highlighting the frames that effectively capture the essential audio-visual information required to answer the given questions.