Technical Summary: MovieChat: From Dense Token to Sparse Memory for Long Video Understanding

Kai-Yu Lu

2025/11/1

1 Research Problem and Motivation

Long video understanding aims to answer questions and conduct open ended dialogue about videos that span thousands of frames and contain multiple scenes and events. Recent multi modal large language models combine image encoders with large language models to handle visual question answering and captioning, but most existing systems only support short clips with several tens of frames and cannot scale to full movies or episodes.

The central problem studied in this work is how to design a vision language system that can process ultra long videos with more than ten thousand frames while keeping computation and memory costs manageable and still maintaining long range temporal consistency. The paper argues that three factors block current approaches: the quadratic cost of dense self attention over frame tokens, the excessive graphics memory required to store all visual tokens, and the difficulty of preserving long term temporal information when only a small set of frames is processed at once.

Inspired by the Atkinson Shiffrin cognitive memory model, which distinguishes short term and long term memory buffers, the authors propose to reinterpret transformer tokens as memory units and to build a structured memory mechanism that converts dense frame level tokens into a compact sparse representation suitable for long video reasoning.

2 Related Work

2.1 Multi modal large language models

Recent multi modal large language models extend language models with visual encoders through either learned projection layers or dedicated query modules. Representative systems such as Flamingo, BLIP and BLIP 2, MiniGPT 4, and Otter align frozen vision encoders with frozen large language models through cross modal transformers or lightweight adapters. These models demonstrate strong performance on image based tasks such as captioning, visual question answering, optical character recognition, and visual reasoning, and they provide a flexible interface for instruction following and dialogue.

Video centric extensions of this paradigm include VideoChat, Video LLaMA, LLaMA Adapter based models, and Video ChatGPT. These systems process a small set of sampled frames, sometimes augmented with motion features, and then feed the resulting tokens into a language model. Although they perform well on short video datasets, they usually handle at most several dozen frames during inference and do not explicitly address ultra long videos.

2.2 Long video understanding

Long video understanding has been explored using classical video models and specialised architectures. Prior work uses three dimensional convolutional networks combined with feature banks, object or human centric representations, or temporal aggregation modules to capture long range dynamics. Other approaches such as MIST decompose dense self attention into segment and region selection modules to reduce computational cost on minute long videos.

On the dataset side, most benchmarks focus on event boundary detection, language grounding from audio descriptions, or question answering with relatively short clips or sparse annotations. MovieQA and MovieNet provide movie related benchmarks but do not supply dense captioning and fine grained question answering for very long clips.

2.3 Memory models in vision tasks

Memory based architectures have been widely used in video object segmentation, multi object tracking, visual object tracking, and action understanding. Methods such as XMem and MeMOT maintain explicit memory banks of past features in order to propagate information over long temporal horizons while limiting computation. These works show that carefully designed memory mechanisms can compensate for the limitations of frame by frame processing.

The present work extends this idea to the setting of multi modal large language models by treating visual tokens as memory items and designing coordinated short term and long term memories that summarise ultra long videos into a sparse but informative representation.

3 Dataset Construction

3.1 Overview of MovieChat 1K

To evaluate long video understanding, the paper constructs MovieChat 1K, a benchmark consisting of 1000 long video clips collected from movies and television series. Each video contains multiple alternating scenes and belongs to one of 15 popular genres such as documentary film, animation film, detective film, epic film, action film, and family film.

The videos have lengths mostly between ten thousand and twelve thousand frames. More than ninety percent of the clips fall in this range, and a non trivial portion exceeds twelve thousand frames, while only a small minority has fewer than ten thousand frames. This design explicitly targets the ultra long regime that existing systems cannot handle in a single pass.

3.2 Annotations and supervision

Each video in MovieChat 1K is annotated with:

- One dense caption that describes the entire clip in natural language, usually containing between one hundred and one hundred and fifty words.
- Three global question answer pairs, where questions refer to the overall content of the video, such as the main event, the outcome, or the high level storyline.
- Ten breakpoint question answer pairs, each associated with a specific timestamp. These questions focus on local events near the breakpoint, for example actions, objects, or interactions in a particular scene.

The questions cover a range of wh question types, including “What”, “Where”, “When”, “How many”, and also yes or no style queries that begin with verbs such as “Is” or “Does”. The majority of questions are open ended, and only about one quarter are multiple choice.

3.3 Dataset statistics

The paper analyses the distribution of question and answer lengths for both global and breakpoint questions. Most questions contain between five and fifteen words, indicating that they are concise yet informative. Answers are usually shorter and often have fewer than ten words, which reflects the open ended but focused nature of the tasks. The dense captions are longer and provide a narrative level summary of each video.

4 Query Protocol and Task Definitions

4.1 Global mode

Global mode evaluates a system’s ability to understand an entire long video. In this mode, the model receives the full video as input along with a global question that may depend on information distributed across many scenes, such as the main topic of a documentary or the resolution of a conflict in a movie. The model must generate a free form text answer.

In MovieChat, global mode uses only the long term memory as the video representation, which encodes a compressed summary of the full clip.

4.2 Breakpoint mode

Breakpoint mode tests localised understanding at specific time points within the video. Each question is associated with a timestamp, and the system must answer based on the visual context around that moment. Examples include identifying what an actor is doing, which objects are currently visible, or how a scene has changed relative to earlier frames.

In MovieChat, breakpoint mode uses a combination of three components:

- The long term memory, which stores condensed information about the entire past.
- The current short term memory, which stores dense tokens for the most recent frames.
- The current frame feature at the query time.

Concatenating these components yields a representation that captures both local details and long range context.

4.3 Task types and evaluation metrics

The benchmark defines two primary task families:

- **Question answering for short videos**: evaluated on existing datasets MSVD QA, MSRVTT QA, and ActivityNet QA. The model predicts text answers given relatively short clips.
- **Question answering for long videos**: evaluated on MovieChat 1K in both global and breakpoint modes.

Evaluation uses two complementary metric types.

1. **Accuracy**: an automatic judgement of whether the predicted answer matches the ground truth answer. The paper employs large language models as evaluators to compare predicted and reference answers and returns a True or False decision.

2. **Relative quality score**: a score on a discrete scale from zero to five that measures how close the prediction is to the correct answer in meaning. For generation quality, additional criteria are used, including correctness of information, detail orientation, contextual understanding, temporal understanding, and consistency. Each criterion is scored between one and five, where higher values denote better performance.

For long video experiments, evaluations are conducted with multiple language model assistants (for example GPT 3.5 and Claude) and further refined through human blind rating. Results with inconsistent automatic judgements and scores are manually filtered to improve reliability.

5 Modeling Approach

5.1 Overall architecture

MovieChat is a multi modal dialogue system that combines:

- A frame wise visual feature extractor that maps raw video frames to visual tokens.
- A short term memory that stores dense tokens for recent frames.
- A long term memory that stores sparse, consolidated tokens for the entire video history.
- A projection module, including a Q former and a linear layer, that converts visual representations into the embedding space of a large language model.
- A large language model that receives both the video representation and the textual question and outputs natural language answers.

The design goal is to transform a dense sequence of frame tokens into a compact sparse memory that can fit within the context limits and memory budget of the language model while still preserving enough information for long range reasoning.

5.2 Visual feature extraction

Let a raw video be denoted by $\mathbf{v} \in \mathbb{Z}^{T\times 3\times H\times W}$, which is a sequence of T RGB frames of spatial size $H \times W$. Each frame is first passed through an image based encoder, such as a vision transformer pretrained in a contrastive manner. The encoder outputs a set of patch tokens for each frame.

To process such a long sequence, MovieChat adopts a non overlapping sliding window strategy over frames. Let C denote the window length measured in frames, and let $V(\cdot)$ denote the visual feature extractor. The n th clip feature within the sliding window can be written as

$$B_n = \{\mathbf{x}_i = V(\mathbf{v}_i) \mid i = 1, \ldots, C\}, \quad n = 1, \ldots, \left\lceil \frac{T}{C} \right\rceil . \tag{1}$$

In Equation (1), B_n is the set of token features for the n th window. Each frame $\mathbf{v}_i$ is mapped by $V(\cdot)$ to a matrix of size $N \times D$, where N is the number of tokens per frame (for example the number of patches) and D is the token dimension. Conceptually, $V(\cdot)$ transforms each image into a collection of semantic "words" that will later act as memory units. The sliding window ensures that the video is processed in manageable chunks rather than as a single enormous sequence.

5.3 Short term memory

Short term memory stores tokens from a limited range of recent frames and acts as a buffer before information is consolidated into long term memory. Suppose that G sliding windows are accumulated in the buffer. The short term memory $\mathcal{S}$ is then defined as the union of the clip features:

$$\mathcal{S} = \bigcup_{n} B_n = \{\mathbf{x}_i \mid i = 1, \ldots, K\}, \quad n = 1, \ldots, G, \tag{2}$$

where $K = C \times G$ is the total number of frames currently stored in the short term memory.

In Equation (2), $\mathcal{S}$ collects all dense frame tokens in a first in first out buffer. The role of short term memory is to provide detailed local context for recent frames, similar to how human short term memory retains detailed information for a limited duration. Once the buffer reaches its maximum capacity, the oldest frames are sent to the consolidation module, and the short term memory is reinitialised using the consolidated representation so that context is propagated between windows.

In the implementation, the sliding window size is set to sixteen frames. The short term memory stores eighteen frames, and each frame contributes thirty two tokens to the memory buffer. This means that short term memory contains detailed information about a local temporal span while remaining bounded in size.

5.4 Long term memory and memory consolidation

Long term memory is responsible for preserving information over the entire video without storing all dense tokens. Directly appending all tokens from short term memory into a long term buffer would cause memory usage to grow linearly with video length and quickly exceed the hardware limit. At the same time, neighbouring frames often exhibit high visual redundancy, since many scenes change slowly.

To address this, MovieChat introduces a memory consolidation algorithm that merges highly similar adjacent frames in token space. For two consecutive frames in the short term memory, the similarity score s is computed as the average cosine similarity over their corresponding tokens:

$$s = \frac{1}{N} \sum_{j=1}^{N} \cos\left(\mathbf{x}_i^{(j)}, \mathbf{x}_{i+1}^{(j)}\right). \tag{3}$$

In Equation (3), $\mathbf{x}_i^{(j)}$ denotes the j th token vector of frame i, and $\mathbf{x}_{i+1}^{(j)}$ denotes the corresponding token of the next frame. The cosine function computes the similarity between two vectors by measuring the angle between them. Averaging over all tokens produces a single similarity value for the frame pair. Intuitively, if successive frames show very similar content, their tokens will be almost parallel in the embedding space, leading to a high cosine similarity, and therefore a high value of s.

The consolidation procedure repeatedly finds the pair of adjacent frames with the largest similarity score and merges their tokens through a weighted average. This process is iterated until only R_L frames remain in the consolidated sequence, where R_L is a hyperparameter that controls the target length of long term memory for each consolidation step. The result is a compressed sequence that preserves salient changes while discarding redundant frames. This compressed sequence is then appended to the long term memory buffer.

The long term memory stores at most 256 frames after consolidation. Because the number of stored frames can exceed the positional encoding length of the original transformer, MovieChat adopts a hierarchically decomposed positional encoding scheme similar to the one proposed for BERT. This scheme extends effective positional coverage from length n to length n^2 by representing positions as combinations of multiple base encodings, which allows the model to assign distinct positions to many more tokens without retraining the encoder.

Description	Default value
Sliding window size	16 frames per window
Short term memory size	18 frames, each with 32 tokens
Long term memory size	256 frames after consolidation
Consolidation length	2 frames merged into one representative frame per operation

Table 1: Hyperparameter settings of the MovieChat memory mechanism.

5.5 Inference modes and language interface

Once the visual tokens have been processed by the memory mechanism, they are passed through a Q former and a linear projection layer that adapt the visual features to the input space of the language model. Let V denote the final video representation produced by either global mode or breakpoint mode. Let $P(\cdot)$ denote the projection from visual space to the language model embedding space, let Q denote the textual question, and let $O(\cdot)$ denote the large language model. The answer $\mathbf{A}$ is produced as

$$\mathbf{A} = O\big(Q, P(V)\big). \tag{4}$$

In Equation (4), $P(V)$ contains a fixed number of visual tokens summarising the video, and Q is a sequence of text tokens encoding the question and dialogue history. The large language model $O(\cdot)$ attends jointly to the question and the projected visual tokens and generates an answer $\mathbf{A}$ in natural language. Conceptually, Equation (4) plays the role of a conditional generator that produces text responses grounded in visual context.

For global mode, V is set equal to the long term memory alone. For breakpoint mode, V is formed by concatenating the long term memory, the current short term memory, and the current frame feature at the query time. This design allows the system to adaptively focus on global context or local details depending on the type of query.

5.6 Hyperparameter configuration

The main hyperparameters of MovieChat are summarised in Table 1. They govern the trade off between accuracy and efficiency.

Ablation studies show that large deviations from these settings reduce performance, which indicates that the chosen configuration balances the need to retain information against the need to keep the token budget manageable.

6 Empirical Results

6.1 Short video question answering

Although MovieChat is designed for long videos, it is also evaluated on standard short video question answering benchmarks. On MSVD QA, MSRVTT QA, and ActivityNet QA, MovieChat achieves higher or comparable accuracy and relative scores compared with baselines such as FrozenBiLM, VideoChat, LLaMA Adapter, Video LLaMA, and Video ChatGPT.

For example, on MSVD QA the accuracy of MovieChat is reported as approximately seventy five percent, substantially higher than the second best baseline. On ActivityNet QA, MovieChat similarly improves both accuracy and quality score. This indicates that the memory mechanism and visual representation are competitive even when the videos are short.

Method	Frames	Global Acc.	Global Score	Breakpoint Acc.	Breakpoint Score
VideoChat	32	57.8	3.00	46.1	2.29
Video LLaMA	32	51.7	2.67	39.1	2.04
Video ChatGPT	100	47.6	2.55	48.0	2.45
MovieChat	2048	62.3	3.23	48.3	2.57

Table 2: Long video question answering performance on MovieChat 1K. Accuracy and relative scores are averaged over multiple language model evaluators and human blind ratings.

6.2 Short video generative performance

To assess generation quality, the paper uses a large language model as an evaluator to assign scores from one to five for several aspects: correctness of information, detail orientation, contextual understanding, temporal understanding, and consistency. On processed ActivityNet QA, MovieChat obtains the highest scores among the compared methods across most criteria.

For instance, MovieChat achieves higher correctness and contextual understanding scores than VideoChat, LLaMA Adapter, and Video LLaMA, which shows that its answers are both more accurate and better grounded in the video content. Temporal understanding and consistency scores are also improved, reflecting stronger modelling of event sequences.

6.3 Long video question answering on MovieChat 1K

The primary evaluation focuses on long video question answering using MovieChat 1K. Baseline methods such as VideoChat, Video LLaMA, and Video ChatGPT are limited to processing at most a small number of frames (for example thirty two or one hundred), so they must subsample the original videos heavily. In contrast, MovieChat processes up to two thousand and forty eight frames while maintaining feasible memory usage.

Table 2 summarises the results in global and breakpoint modes. MovieChat achieves the highest global accuracy and global quality score and performs competitively in breakpoint mode.

The improvements in global mode mean that MovieChat better captures the overall narrative of the long videos. The gains in score also indicate that its answers are judged to be of higher semantic quality and more faithful to the reference answers.

6.4 Long video generative performance

For long video generation quality in global mode, MovieChat again obtains the best or second best scores across all criteria. It reaches the highest scores in correctness of information, detail orientation, contextual understanding, and temporal understanding, and also achieves a strong consistency score. This suggests that the memory mechanism allows the model to produce text that is coherent with long range temporal structure and rich in relevant details.

6.5 Ablation studies

The paper conducts ablation studies to understand the contribution of the memory mechanism and the effect of hyperparameters.

- **Effect of memory mechanism**: Removing the memory mechanism and directly using dense tokens leads to a significant drop in both accuracy and generation quality for long video question answering.

This confirms that explicitly structured short term and long term memories are essential for scaling to ultra long videos.

- **Effect of buffer lengths**: Varying the lengths of short term and long term memory buffers shows that performance improves as these buffers grow, up to a point, because more visual information is retained. However, very large buffers cause information to be overly compressed in consolidation, which harms fine detail.

- **Effect of consolidation length and initialisation**: The consolidation length controls how aggressively adjacent frames are merged. Too small a value yields limited compression, whereas too large a value merges frames that are not sufficiently similar and destroys temporal detail. The ablation shows that merging two frames at a time is a good compromise. For the initialisation of the short term memory after consolidation, using merged tokens performs better than using only the last few dense tokens or using uniformly sampled tokens, which indicates that the merged representation better captures the content of the previous window.

7 Summary

This work introduces MovieChat, a multi modal dialogue system for long video understanding, and MovieChat 1K, a benchmark dataset designed to evaluate this setting. The key technical contribution is a memory mechanism that converts dense frame level tokens into sparse memories through sliding window extraction, short term buffering, and similarity based consolidation into long term memory. This mechanism enables the system to handle videos with more than ten thousand frames while controlling memory usage and preserving long range temporal information.

Experiments show that MovieChat matches or surpasses existing video centric multi modal language models on standard short video benchmarks and achieves superior performance on long video question answering and generation tasks. Ablation studies highlight the importance of both short term and long term memory buffers and validate the chosen hyperparameter configuration.

The paper also identifies limitations, including dependence on the quality of pretrained short video encoders and relatively coarse modelling of precise temporal durations. Future work may focus on integrating stronger video foundation models, refining temporal reasoning with more precise time representations, and extending the approach to broader domains such as instructional videos and egocentric recordings.

Key takeaways.

1. Long video understanding requires explicit mechanisms to manage computational and memory cost while preserving long range temporal information, which is not addressed by most existing multi modal language models.

2. The proposed short term and long term memory mechanism, based on sliding window extraction and similarity based token consolidation, provides an effective way to compress dense frame tokens into a compact sparse memory that still supports detailed reasoning.

3. The MovieChat system and MovieChat 1K benchmark demonstrate that such a memory centric design can achieve state of the art performance on ultra long video question answering, both in terms of automatic accuracy and human judged generation quality.