HiMemFormer: Hierarchical Memory-Aware Transformer for Multi-Agent Action Anticipation

Zirui Wang[1] **Xinran Zhao**[2] **Simon Stepputtis**[2] **Woojun Kim**[2]
Tongshuang Wu[2] **Katia Sycara**[2] **Yaqi Xie**[2]
[1]University of Illinois at Urbana-Champaign [2]Carnegie Mellon University
`ziruiw3@illinois.edu`
`{xinranz3,sstepput,woojunk,sherryw,sycara,yaqix}@andrew.cmu.edu`

Abstract

Understanding and predicting human actions has been a long-standing challenge and is a crucial measure of perception in robotics AI. While significant progress has been made in anticipating the future actions of individual agents, prior work has largely overlooked a key aspect of real-world human activity – interactions. To address this gap in human-like forecasting within multi-agent environments, we present the Hierarchical Memory-Aware Transformer (HiMemFormer), a transformer-based model for online multi-agent action anticipation. HiMemFormer integrates and distributes global memory that captures joint historical information across all agents through a transformer framework, with a hierarchical local memory decoder that interprets agent-specific features based on these global representations using a coarse-to-fine strategy. In contrast to previous approaches, HiMemFormer uniquely hierarchically applies the global context with agent-specific preferences to avoid noisy or redundant information in multi-agent action anticipation. Extensive experiments on various multi-agent scenarios demonstrate the significant performance of HiMemFormer, compared with other state-of-the-art methods.

1 Introduction

Action detection [8] or anticipation [21] systems aim at forecasting future states of single or multiple agents from history. The recent advances in these areas facilitate embodied or virtual AI systems with the ability to perceive and interact with other agents and complex environments [33, 37, 44]. Such ability plays a pivotal role in numerous applications, such as autonomous driving [41], collaborative robotics [30], and home automation [32], where understanding and predicting the actions of various entities in a shared environment can significantly enhance safety, efficiency, and coordination.

Agent memory plays an important role in conducting action anticipation due to the innate dependencies among actions [39, 36, 14, 37, 44]. LSTR [39] proposes to capture both long-term and short-term memory, while MAT [36] additionally incorporates future content in seen scenarios. In the multi-agent scenarios [33], each agent can be arbitrary or affected by the environment, which suggests one key to the success of a multi-agent system: *how to effectively capture agent behavior at various time and social scales*. A prominent line of research exploits the ways to obtain a unified single global feature representing time, e.g., [23, 4], and social relations, e.g., [17, 29]. AgentFormer [42] and HiVT [45] further explore combining time and social features with a overall global representation.

Despite the significance, these state-of-the-art systems overlook the individual perspective of the problem: different agents may need time and social features at different scales. From the time perspective, some agent actions heavily rely on long-term memory, e.g., if they belong to a complex multi-step action sequence, while some actions are only relevant to short memory, e.g., an instant

Workshop on Video-Language Models at 38th Conference on Neural Information Processing Systems (NeurIPS 2024).

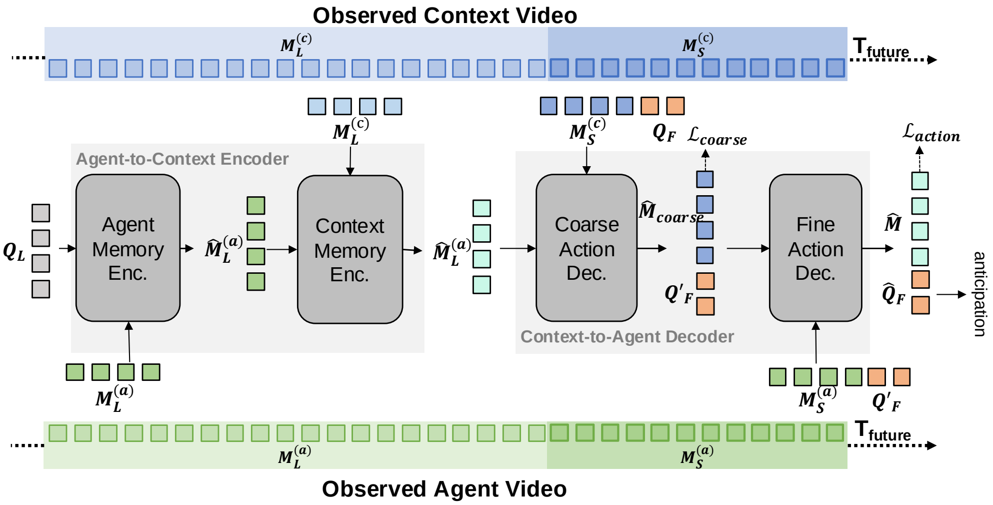

Figure 1: **HiMemFormer Architecture** In the Agent-to-Context Encoder, the observed agent's long-term memory is encoded to a abstract representation $\hat{\mathrm{M}}_L^{(a)}$ and cross-attention with context past history $\hat{\mathrm{M}}_L^{(c)}$. Then, the Context-to-Agent Decoder utilize both agent and global recent memories to learn the future information through a two-stage refinement approach.

response to a rapid environment change. From the social perspective, similarly, the actions of some agents are much correlated with others during collaboration, while some of mostly stand-alone. To capture these agent-specific preferences in feature utilization, we propose to *hierarchically* capture the time and social features for each agent-specific decoder to include these global or contextual features with the desired granularity and discard unnecessary information that may introduce noise or latency to *each specific agent*.

To achieve customized and flexible global feature utilization automatically, we propose the Hierarchical Memory-Aware Transformer (HiMemFormer), a novel approach that simultaneously learns feature representations from both contextual and agent-specified dimensions through a dual-hierarchical framework. Specifically, its Agent-to-Context Encoder augments the agents' long-term history through cross-attention with global long-term memory. Then, the encoded long-term memory is further processed through a hierarchical Agent-to-Context Decoder that offers a coarse prediction given augmented long-term memory and contextual short-term memories. Finally, the coarse prediction is gradually refined by each agent-specific network augmented with individual short-term memory to get the anticipated actions. Through the dual-hierarchical network, HiMemFormer manages to model agent's unique short-term memory while learning useful correlations from the contextual memories. This allows us to effectively compress the long-range contextual information without losing important lower level feature information. In summary, our contributions are three-fold:

- We propose a transformer-based method to capture and utilize the global features in multi-agent scenarios in a flexible way responding to each agent's preference.
- We design a hierarchical memory encoder that follows a specific-to-general paradigm to learn long-term joint-memory features and a hierarchical memory decoder that learns an agent's future action by a coarse-to-fine strategy.
- We carry out exhaustive experiments on various multi-agent action anticipation scenarios and outperform existing baseline models.

2 Hierarchical Memory-Aware Transformer

We consider the general setting of multi-agent action learning as, given target agent's live streaming First-Person-View (FPV) video, along with the Third-Person-View (TPV) video of the whole scene, our goal is to predict individual agent's actions in a time period τ using only past and cur-

Table 1: **Results of online action anticipation on LEMMA** [20] using SlowFast features in up to 2 seconds. In particular, we report accuracy in mAP across 4 scenarios, including single agent scenarios where it perform single or multiple task, and multi-agent scenarios where multiple agents collaborate on single task or carry out separate tasks.

	Scenarios (# Agent × # Task)			
	1×1	1×2	2×1	2×2
LSTR[26]	75.8	50.9	47.0	68.0
MAT [36]	73.0	50.8	50.4	67.1
HiMemFormer (ours)	**76.3**	**54.2**	48.4	**70.6**
HiMemFormer+ (ours)	76.2	52.2	**50.5**	69.9

rent observations. To tackle this problem, we introduce Hierarchical Memory-Aware Transformer (HiMemFormer), a transformer-based model with encoder-decoder architecture, as shown in Fig. 1. In particular, given observed agent's long-term history $\mathbf{M}_L^{(a)}$, we compress it to a latent representation of fixed size through a transformer unit and then cross-attentioned with contextual long-term history $\mathbf{M}_L^{(c)}$to get the final encoded long-term memory $\mathbf{M}_L$. Using a coarse-to-fine strategy, we first decode the long-term memory by cross-attentioned with short-term global memory $\mathbf{M}_S^{(c)}$ to learn all possible actions in the current state, and refine the predicted actions by cross-attention with agent's recent past information $\mathbf{M}_S^{(a)}$ to get the final anticipated action. See details in Appendix B

3 Experiments

3.1 Datasets and Metrics

We evaluate our model on a public-available multi-agent dataset LEMMA [20], which includes 862 compositional atomic-action from 324 activities with 445 egocentric videos (multiple egocentric videos for multi-agent activities). We follow prior work [20] on the dataset split, evaluating four scenarios: single-agent single-task (1×1), single-agent multi-tasks (1×2), multi-agent single-task (2×1)and multi-agent multi-tasks(2×2). For online action anticipation, we follow prior works [20, 10, 39, 36] and evaluate on per-frame mean average precision (mAP) to measure the performance and evaluate over an anticipation period of $\tau_f = 2s$. See details in Appendix C.1.

3.2 Results and Discussion

We compared HiMemFormer with other baseline models [39, 36] on LEMMA [20]. Specifically, we set HiMemFormer with 64 seconds and 5 seconds for long and short-term memories, respectively. Table 1 demonstrates that HiMemFormer significantly outperforms LSTR [39] by at 0.8%, 4%, 1.9% and 0.8% for all four scenarios respectively in terms of mAP, demonstrating the effectiveness of the hierarchical design of joint agent-specific and contextual memory for inferencing. It is worth noting that HiMemFormer also outperforms MAT [36] by a larger margin in 2×2 scenario, given that MAT additionally learns features from the future. This critical observation indicates the significance of hierarchical global information in multi-agent action anticipation. To ensure a fair comparison, we develop HiMemFormer+ on top of MAT [36] to align with its temporal feature on utilizing extra future features, and integrate the hierarchical transformer block for multi-agent action anticipation. Results shows around 2% improvements over both baselines. More details in Appendix C.3 and C.4.

4 Conclusion

We present Hierarchical Memory-Aware Transformer (HiMemFormer), a transformer-based architecture with hierachical global and local memory attention mechanisms for online action anticipation, to overcome the weakness of the existing methods that can only complete modeling temporal dependency or only modeling agent interaction dependency without considering global historical context. Through experiments on four different scenarios involving multi-agents interactions, we show its capability of modeling both temporal and spatial dependencies, demonstrating the importance of both long-term historical context and short-term agent-specific information.

References

[1] Y. Abu Farha and J. Gall. Uncertainty-aware anticipation of activities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[2] Y. Abu Farha, A. Richard, and J. Gall. When will you do what?-anticipating temporal occurrences of activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5343–5352, 2018.

[3] Y. Abu Farha, Q. Ke, B. Schiele, and J. Gall. Long-term anticipation of activities with cycle consistency. In *Pattern Recognition: 42nd DAGM German Conference, DAGM GCPR 2020, Tübingen, Germany, September 28–October 1, 2020, Proceedings 42*, pages 159–173. Springer, 2021.

[4] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[5] S. Bhagat, S. Stepputtis, J. Campbell, and K. Sycara. Knowledge-guided short-context action anticipation in human-centric videos, 2023. URL `https://arxiv.org/abs/2309.05943`.

[6] J. Chen, Z. Lv, S. Wu, K. Q. Lin, C. Song, D. Gao, J.-W. Liu, Z. Gao, D. Mao, and M. Z. Shou. Videollm-online: Online video large language model for streaming video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18407–18418, 2024.

[7] D. Damen, H. Doughty, G. M. Farinella, A. Furnari, J. Ma, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. URL `https://doi.org/10.1007/s11263-021-01531-2`.

[8] R. De Geest, E. Gavves, A. Ghodrati, Z. Li, C. Snoek, and T. Tuytelaars. Online action detection. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 269–284. Springer, 2016.

[9] H. Fan, Y. Li, B. Xiong, W.-Y. Lo, and C. Feichtenhofer. Pyslowfast. `https://github.com/facebookresearch/slowfast`, 2020.

[10] A. Furnari and G. M. Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 6252–6261, 2019.

[11] A. Furnari, S. Battiato, and G. Maria Farinella. Leveraging uncertainty to rethink loss functions and evaluation measures for egocentric action anticipation. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.

[12] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes. Predicting the future: A jointly learnt model for action anticipation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5562–5571, 2019.

[13] H. Girase, N. Agarwal, C. Choi, and K. Mangalam. Latency matters: Real-time action forecasting transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18759–18769, 2023.

[14] D. Gong, J. Lee, M. Kim, S. J. Ha, and M. Cho. Future transformer for long-term action anticipation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3052–3061, 2022.

[15] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S. K. Ramakrishnan, F. Ryan, J. Sharma, M. Wray, M. Xu, E. Z. Xu, C. Zhao, S. Bansal, D. Batra, V. Cartillier, S. Crane, T. Do, M. Doulaty, A. Erapalli, C. Feichtenhofer, A. Fragomeni, Q. Fu, C. Fuegen, A. Gebreselasie, C. Gonzalez, J. Hillis, X. Huang, Y. Huang, W. Jia, W. Khoo, J. Kolar, S. Kottur, A. Kumar, F. Landini, C. Li, Y. Li, Z. Li, K. Mangalam, R. Modhugu, J. Munro, T. Murrell, T. Nishiyasu,

W. Price, P. R. Puentes, M. Ramazanova, L. Sari, K. Somasundaram, A. Southerland, Y. Sugano, R. Tao, M. Vo, Y. Wang, X. Wu, T. Yagi, Y. Zhu, P. Arbelaez, D. Crandall, D. Damen, G. M. Farinella, B. Ghanem, V. K. Ithapu, C. V. Jawahar, H. Joo, K. Kitani, H. Li, R. Newcombe, A. Oliva, H. S. Park, J. M. Rehg, Y. Sato, J. Shi, M. Z. Shou, A. Torralba, L. Torresani, M. Yan, and J. Malik. Ego4d: Around the World in 3,000 Hours of Egocentric Video. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2022.

[16] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, et al. Knowledge graphs. *ACM Computing Surveys (Csur)*, 54 (4):1–37, 2021.

[17] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6272–6281, 2019.

[18] Y. Huang, X. Yang, and C. Xu. Multimodal global relation knowledge distillation for egocentric action anticipation. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 245–254, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450386517. doi: 10.1145/3474085.3475327. URL `https://doi.org/10.1145/3474085.3475327`.

[19] A. Jain, A. Singh, H. S. Koppula, S. Soh, and A. Saxena. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3118–3125. IEEE, 2016.

[20] B. Jia, Y. Chen, S. Huang, Y. Zhu, and S.-C. Zhu. Lemma: A multiview dataset for learning multi-agent multi-view activities. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[21] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert. Activity forecasting. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part IV 12*, pages 201–214. Springer, 2012.

[22] Y. Kong, S. Gao, B. Sun, and Y. Fu. Action prediction from videos via memorizing hard-to-predict samples. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[23] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *Advances in neural information processing systems*, 32, 2019.

[24] Y. Li, M. Liu, and J. M. Rehg. In the eye of the beholder: Gaze and actions in first person video. *IEEE transactions on pattern analysis and machine intelligence*, 45(6):6731–6747, 2021.

[25] Y. Li, C.-Y. Wu, H. Fan, K. Mangalam, B. Xiong, J. Malik, and C. Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4804–4814, 2022.

[26] S. B. Loh, D. Roy, and B. Fernando. Long-term action forecasting using multi-headed attention-based variational recurrent neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2419–2427, 2022.

[27] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[28] C. Rodriguez, B. Fernando, and H. Li. Action anticipation by predicting future dynamic images. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[29] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 683–700. Springer, 2020.

[30] P. Schydlo, M. Rakovic, L. Jamone, and J. Santos-Victor. Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action sequences prediction. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5909–5914. IEEE, 2018.

[31] F. Sener, D. Singhania, and A. Yao. Temporal aggregate representations for long-range video understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 154–171. Springer, 2020.

[32] B. Soran, A. Farhadi, and L. Shapiro. Generating notifications for missing actions: Don't forget to turn the lights off! In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4669–4677, 2015.

[33] K. P. Sycara. Multiagent systems. *AI magazine*, 19(2):79–79, 1998.

[34] Z. Tong, Y. Song, J. Wang, and L. Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[36] J. Wang, G. Chen, Y. Huang, L. Wang, and T. Lu. Memory-and-anticipation transformer for online action understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13824–13835, 2023.

[37] C.-Y. Wu, Y. Li, K. Mangalam, H. Fan, B. Xiong, J. Malik, and C. Feichtenhofer. MeMViT: Memory-Augmented Multiscale Vision Transformer for Efficient Long-Term Video Recognition. In *CVPR*, 2022.

[38] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017.

[39] M. Xu, Y. Xiong, H. Chen, X. Li, W. Xia, Z. Tu, and S. Soatto. Long short-term transformer for online action detection. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[40] L. Yang, J. Han, and D. Zhang. Colar: Effective and efficient online action detection by consulting exemplars. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3160–3169, 2022.

[41] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 507–523. Springer, 2020.

[42] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021.

[43] Q. Zhao, C. Zhang, S. Wang, C. Fu, N. Agarwal, K. Lee, and C. Sun. Antgpt: Can large language models help long-term action anticipation from videos? *arXiv preprint arXiv:2307.16368*, 2023.

[44] Y. Zhao and P. Krähenbühl. Real-time online video detection with temporal smoothing transformers. In *European Conference on Computer Vision (ECCV)*, 2022.

[45] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8823–8833, 2022.

[46] J. Zou, M. Zhou, T. Li, S. Han, and D. Zhang. Promptintern: Saving inference costs by internalizing recurrent prompt during large language model fine-tuning. *arXiv preprint arXiv:2407.02211*, 2024.

A Related Work

Action Anticipation Online action anticipation [21] aims to predict the future actions of the agent given the past and current action information. Given its' increasing popularity and its broad practical applications, many large-scale datasets and benchmarks [7, 15, 24] has been proposed to facilitate researchers in this area. In the field of action anticipation, feature learning [28, 12] and temporal modeling [2, 14] are the two main streams of approaches. Recurrent neural network (RNN) is widely adopted by many prior works [1, 3, 12, 22, 26, 10] due to its powerful long-term temporal dependency. For example, RULSTM [10] proposed to anticipate actions via a rolling LSTM to encode historical information and unrolling LSTM make predictions on future actions. To better model time and social dimension in a multi-agent setting, a popular line of research [23, 4] uses temporal models to summarize features over time for each agent separately and then feed temporal features into social models to obtain global-aware agent features. There are also works [17, 29] that uses social models to generate social features of for individual agents and apply temporal models to summarize social features for each agent. For example, Trajection++ [29] design a graph-structured recurrent model that forecasts the trajectories of a general number of diverse agents while incorporating agent dynamics and heterogeneous data. However, previous methods failed to consider both temporal dependencies and social dependencies at once, which can be sub-optimal. More recent work [42, 45] manage to overcome this short coming by considering both time and social dimensions simultaneously, facilitating interaction across temporal domain and spatial domain.

Transformer for Video Understanding Recently, transformer-based methods [14, 36, 39, 25, 34, 40, 13] stood out in literature because of its strong capability for long-range temporal dependencies. For example, Gong *et al.* [14] proposed Future Transformer (FUTR), an end-to-end attention neural network that anticipate actions in parallel decoding, leveraging global interactions between past and future actions for long-term action anticipation. LSTR [39] further decomposes the memory encoder into long and short-term stages for online action detection and anticipation, allowing model to learn more representative features from the history. MAT [36] proposes a new memory-anticipation-based paradigm that models the entire temporal structure, including past, present and future. Also to utilize different sources such as optical flow and audio data, multi-modal fusion approaches [10, 11, 19, 31] has been proposed to improve the accuracy of future action prediction. In addition, large language model (LLM) [43] is deployed to tackle action anticipation task due to its strong high-level reasoning capability.

B HiMemFormer Details

B.1 Agent-to-Context Encoder

Agent-specific long-term memory provide useful information about the historical actions of the agent, but when placed in a complex environment with multi-agent interactions, it is crucial to pay attention to contextual information that are shared across all agents. To this end, Agent-to-Context Memory Encoder follows the specific-to-general approach, managing to augment agent's long-term memory by paying extra attention to global features via cross-attention.

Agent Memory Encoding. For each agent in a scene, we input target agent's long-term memory features $\mathbf{M}_L^{(a)}$ to the Transformer Block and compress target agent's long-term feature into a latent representation of fixed length. Following prior work [39, 44], we utilize a two-stage memory compression transformer module, denoted $\mathcal{F}_L^{(a)}$, consists of multiple transformer decoder unit [35] to encoder the agent-specific long-term history $\widehat{\mathbf{M}}_L^{(a)}$:

$$\widehat{\mathbf{M}}_L^{(a)} = \mathcal{F}_L^{(a)}(\mathbf{M}_L^{(a)}, \mathbf{M}_L^{(a)}) \tag{1}$$

Context Memory Enhancement. To effectively encode contextual information into agent's long-term memory, we propose a specific-to-general approach. In practice, we send contextual long-term history $\mathbf{M}_L^{(c)}$ (with positional embedding) as queries and $\widehat{\mathbf{M}}_L^{(a)}$ to our context encoder, $\mathcal{F}_L^{(c)}$, constructed with a transformer decoder architecture [35]. Using contextual long-term history to guide the encoded agent's long-term history, we acquire the final encoded long-term memory $\widehat{\mathbf{M}}_L$:

$$\widehat{\mathbf{M}}_L = \mathcal{F}_L^{(c)}(\mathbf{M}_L^{(c)}, \widehat{\mathbf{M}}_L^{(a)}) \tag{2}$$

where inputs to the decoder $\mathcal{F}$ are queries, and key/value pairs.

B.2 Context-to-Agent Decoder

To implement our key idea to predict agent future actions based on both contextual information and agent-specific information, we meticulously design our Context-to-Agent Decoder using a coarse-to-fine apporach. In particular, leveraging informative short-term features, as demonstrated in LSTR [39], we first make a coarse prediction that contains possible future actions of all agents in the scene using contextual short-term features, and narrow down to target agent's future action using agent-specific short-term features as queries to the transformers units. We also supervise on both coarse and precise action predictions.

Coarse Action Anticipation To enable the model to learn future actions, we need to generate a latent embedding that allows the model to learn future actions from the past and the present. In practise, we initialize N_F learnable query tokens $\mathbf{Q}_F \in \mathbb{R}^{N_F \times D}$ where D is the feature dimension and concatenate with contextual short-term memories $\mathbf{M}_S^{(c)}$ to form $\mathbf{M}_{coarse} = \{\mathbf{M}_S^{(c)}, \mathbf{Q}_F^{(c)}\}$. We then take $\mathbf{M}_{coarse}^{(c)}$ as queries and cross-attentioned with augmented long-term memory $\mathbf{M}_L$ through a transformer decoder architecture [35] $\mathcal{F}_{coarse}$ to make coarse action predictions given only contextual information:

$$\widehat{\mathbf{M}}_{coarse} = \mathcal{F}_{coarse}(\mathbf{M}_{coarse}, \widehat{\mathbf{M}}_L) \tag{3}$$

Precise Action Refinement Until now, the future action query token contain general information about agent's future action. To generate more accurate predicted actions, we leverage the agent-specific short-term history, $\mathbf{M}_S^{(a)}$, that contains feature representations of agent's unique feature and concatenate them with learnable query tokens from the coarse prediction $\mathbf{Q}'_F$ to form $\mathbf{M}_{fine} = \{\mathbf{M}_S^{(a)}, \mathbf{Q}'_F\}$. Following the similar recipe we add another transformer block, denoted $\mathcal{F}_{fine}$ to generate the final action prediction $\widehat{\mathbf{M}}$:

$$\widehat{\mathbf{M}} = \mathcal{F}_{fine}(\mathbf{M}_{fine}, \widehat{\mathbf{M}}_{coarse}) \tag{4}$$

B.3 Training Objectives

The loss function for HiMemFormer comprises two essential components: the coarse action loss $\mathcal{L}_{coarse}$ and the refined action loss $\mathcal{L}_{fine}$. The overall respresentation is:

$$\mathcal{L} = \lambda_a \cdot \mathcal{L}_{coarse} + \lambda_b \cdot \mathcal{L}_{fine} \tag{5}$$

We then use the empirical cross entropy loss between each agent's predicted action anticipation probability distribution $\hat{P}_t \in \mathbb{R}^{T \times (K+1)}$ and the ground truth anticipation label $y_t \in \{0, 1, ..., K\}$. For the coarse action anticipation, we utilize the ground truth anticipation label of all agents in the scene. For the refined action anticipation, we ony use the ground truth anticipation label of the target agent.

C Experiments

C.1 Data Splits

We randomly split all the video samples into training and test sets with ratio of 3 to 1, resulting in 243 recorded activities for training and 81 for validation. Due to the multi-agent setup, the model

Table 2: **Exploring different short-term memory size** of HiMemFormer. In particular, we fixed the long-term memory size to 64 seconds. M_S represents length of short-term memory (in secs).

	Scenarios (# Agent × # Task)			
	1×1	1×2	2×1	2×2
$M_S = 2$	73.5	52.1	47.8	**73.4**
$M_S = 5$	**76.3**	**54.2**	**48.4**	70.6
$M_S = 10$	76.1	47.5	45.4	68.3

Table 3: **Exploring different long-term memory size** of HiMemFormer. In particular, we fixed the short-term memory size to 5 seconds. M_L represents length of long-term memory (in secs).

	Scenarios (# Agent × # Task)			
	1×1	1×2	2×1	2×2
$M_L = 32$	76.3	53.7	47.7	70
$M_L = 64$	76.3	**54.2**	**48.4**	**70.6**
$M_L = 128$	**77.4**	51.2	45.4	68.2
$M_L = 256$	74.5	48.9	46.4	68.7

will be trained on 333 out of 445 egocentric videos, as each activity may have multiple egocentric videos. For action anticipation task, we follow prior work and split training and validation sets with ratios 3: 1, 1:3, 1: 3 and 1:3 for the four scenarios $1 \times 1, 1 \times 2, 2 \times 1, 2 \times 2$, respectively, resulting in (96,19,16,13) activities for training and (31, 57, 50, 42) activities for evaluation in four scenarios.

C.2 Experiment Settings

We implemented our proposed model in PyTorch [27] and performed all experiments on a system with a single NVIDIA A40 graphics cards. For all transformer blocks inside both encoder and decoder module, we set the number of heads to 4 and hidden units as 1024 dimensions. The model is optimized by AdamW optimizer with a weight decay of 1×10^{-4} We use warm-up learning rate linearly increase from zero to 7×10^{-5} in the first 10 epoch. In addition, the model is optimized with batch size of 16 and training is terminated after 25 epochs. Following prior work's experiment settings, we use a pretrained feature extractor [9] extract action features from the video.

C.3 Comparison with Baselines

We compare HiMemFormer with prior methods on LEMMA for action anticipation in both single- and multi-agent environment. Specifically, both baselines, LSTR [39] and MAT [36], only take in agent's first-person-view live streaming video as the input without considering the contextual information from the third-person-view videos. The above set up ensure that the performance gain shown in table 1 come from the proposed multi-view integration of agent and context memory.

Table 4: **Effect of different down-sampling rate for long-term memory** [20] on LEMMA [20]. In particularly, we implement HiMemForMer with long-term memory size of 64 seconds and short-term memory size of 5 seconds.

	Scenarios (# Agent × # Task)			
Sampling Rate (SR)	1×1	1×2	2×1	2×2
$SR = 1$	77.4	52.7	49.3	69.7
$SR = 4$	76.3	54.2	48.4	70.6
$SR = 8$	76.3	53.3	49.2	69.6
$SR = 16$	73.0	52.3	47.0	67.6

C.4 Ablation Studies

Effect of Memory Size We first analyze the effect of the length of both short term and long term memory. Following LSTR [39], we fix the short-term memory to 5 seconds and test $M_L \in \{32, 64, 128, 256\}$, while maintaining same memory size for multi-view videos. Similarly, we fix the long-term memory to 64 seconds and test $M_S \in \{2, 5, 10\}$. Results shown in table 2 and 3 reach the same conclusion in [39], where increasing memory size does not always guarantee better performance.

Effect of Down-Sampling Rate We also test the effect of compression ratio of long-term memory, and we implement HiMemFormer with 5 seconds of short-term memories (both agent and global perspective) and 64 seconds of long-term memories. Results are shown in table 4.

C.5 Discussion on Future Directions

HiMemFormer serves as an attempt to tackle action-anticipation in complex multi-agent environment, but there's more to be explored. Future work could focus on expanding HiMemFormer's capabilities to better interpret complex multi-agent interactions. Potential directions include leveraging large language models (LLMs) to enhance model's interpretability and flexibility in the dynamic environment [43, 6, 46]. Knowledge graphs [16] or scene graphs [38] can also serves as a powerful feature representations [5, 18]to further boost the performance. Additionally, while this study demonstrates the importance of long-term historical context and short-term agent-specific information, exploring adaptive mechanisms that dynamically adjust the emphasis between these dependencies could further improve the model's responsiveness to real-time changes.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

   Answer: [Yes]

   Justification: The abstract is constructed based on the main contribution and scope of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [No]

Justification: Limitation is not discussed in the paper.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper does not include any theoretical results so there's no proof attached.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper explains implementation details in the appendix and also explains the architecture in the main paper.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Currently we are not planning to release the code.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental Settings and detailed information about design choice are listed in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The experiment in the paper is a proof of concept and further experiment on statistical significance will be carried out in the future development.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details of hardware compute resources is mentioned in the appendix of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: I have reviewed the Code of Ethics and strictly follow the rules.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have mentioned in the introduction section where the current work has great potential in the field of robotics.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

Justification: the paper does not posese such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: The paper and its codebase is developed from scratch by the author without using existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not relase new assests so it's not applicable.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: the paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.