MLLM Frame Subset Ensembling for Audio-Visual Video QA and MLLM-based Reranking for Ad-hoc Video Search in TRECVID 2025

Andreas Goulas[1,2], Damianos Galanopoulos[1], Ioannis Patras[2], and Vasileios Mezaris[1]

[1]Information Technologies Institute (ITI), Centre of Research and Technology Hellas (CERTH), Thessaloniki, Greece
[2]Queen Mary University of London, London, UK
{agoulas, dgalanop, bmezaris}@iti.gr, i.patras@qmul.ac.uk

Abstract

This paper presents the overview of the runs related to the Ad-hoc Video Search (AVS) and Video Question Answering (VQA) tracks of TRECVID 2025 on behalf of the CERTH-ITI team. For the AVS track, we introduce a two-stage framework built on foundation models. In the first stage, multiple vision–language models (VLMs) encode both the input query, augmented through LLM-generated rephrasings, and the candidate video shots, producing weighted similarity scores for initial retrieval. In the second stage, we utilize a Multimodal-LLM(MLLM)-based reranking module that evaluates the semantic alignment between each shot among the top-N highest-ranked ones and the original query, generating updated relevance scores for reordering these shots. This MLLM-driven reranking significantly improves contextual matching and produces more accurate final rankings without requiring any model training. Regarding the VQA track, we fine-tune an audio-visual MLLM model on the provided TRECVID training dataset and we implement an inference-time scaling technique to enhance the multimodal understanding capabilities of the MLLM. For the open-ended Answer Generation (AG) task, we aggregate multiple model responses per question via a majority vote. The responses are generated with greedy sampling from different random frame subsets of the video and they are ranked based on the number of votes. For the Multiple-Choice (MC) task, instead of voting, we use mean pooling on the logits assigned by the fine-tuned model to each candidate response. Through the combination of fine-tuning and frame subset ensembling we achieve the highest score across 3 metrics in the VQA AG task and the second highest in the VQA MC task.

1 Introduction

CERTH-ITI has been participating in TRECVID [1], either in collaboration with other organizations under the umbrella of a specific EU-funded project, or as a stand-alone organization, since 2004. In this paper, we present the CERTH-ITI approaches for the Multimedia Tracks of TREC 2025 [2], specifically the Ad-hoc Video Search (AVS) [3] and the Video Question Answering (VQA) [4] tracks. The goal of an AVS method is to retrieve relevant video shots using free-text queries. On the other hand, a VQA method focuses on answering questions related to a specific video, either in an open-ended format or through multiple-choice options. The recent advances in foundation models enable more robust and semantically aligned representations across text and visual domains. This year, we built upon such foundation models to create reliable and well-performing methods on both tracks. For the AVS track, our approach focuses on utilizing foundation models to improve text–video retrieval performance. Building upon our previous submissions [5, 6], we adopt a fully training-free pipeline that integrates multiple pre-trained Vision–Language Models (VLMs) for video shot retrieval, and Large

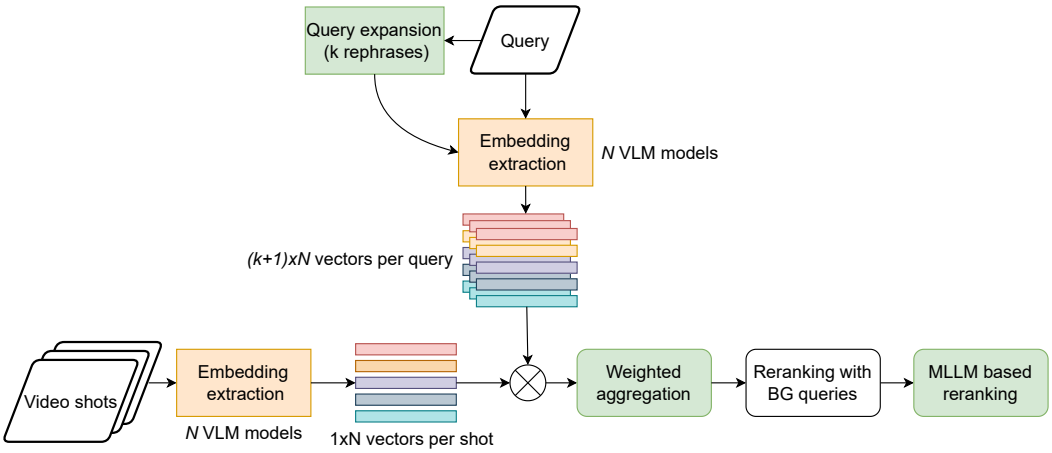

Figure 1: An overview of the proposed AVS method.

Language Models (LLMs) for query augmentation and contextual re-ranking. For the VQA track, we fine-tune a pre-trained audio-visual MLLM to address the two VQA tasks and, during inference, we scale the capabilities of the model by aggregating multiple predictions from the model, generated with greedy sampling. Specifically, for each generation the model sees a different random subset of the video frames. The following sections will present the employed algorithms and the evaluation of our runs in the AVS and VQA tracks, respectively.

2 Ad-hoc Video Search

The TRECVID 2025 [2] Ad-hoc Video Search (AVS) task aims to develop systems capable of retrieving a ranked list of 1,000 video shots for each free-form textual query, ordered from the most to the least relevant. Building on our previous participations, we address this task by leveraging a diverse set of pre-trained image-text models to enhance AVS performance. Moreover, the recent rise of multimodal large language models (MLLMs) [7, 8] has expanded the possibilities for text-based video retrieval. Motivated by these developments, we propose a new two-stage framework that integrates multiple VLMs for image-based retrieval, an LLM for query augmentation and an MLLM for contextual re-ranking. In contrast to our earlier approaches [6, 5], which combined outputs of several cross-modal networks through a trainable neural aggregation module, this year's system does not involve any training. Our internal experiments showed that additional training provided only marginal gains while requiring substantial computational resources. To balance performance and efficiency, we adopt a fixed weighting scheme. Furthermore, we incorporate two LLMs: one for augmenting the input queries and an MLLM to rerank the retrieved video shots by producing contextual relevance scores that assess how well each shot matches the query.

2.1 Approach

In our AVS 2025 submission, we build upon and extend our AVS 2024 approach by introducing a refined two-stage retrieval–reranking framework. The complete AVS architecture is shown in Figure 1. In the retrieval stage, the system analyzes the user query, generates multiple query augmentations, extracts VLM-based embeddings, computes similarities between the (augmented) query and all candidate videos, and retrieves the most relevant video shots. In the reranking stage, we leverage an MLLM to reassess and refine these retrieval results by evaluating how well each retrieved video matches the original query.

In the retrieval stage, in order to enrich the semantic representation of the input query q, we generate multiple variations that reflect different levels of textual detail and help reveal implicit information. For this step, we employ *LLaMA-3.2-1B-Instruct* [9, 10], prompting it to rephrase the

Table 1: The prompts that were used for query augmentation and for the video shot reranking.

LLM	Prompts
LLaMA-3.2-1B-Instruct (query expansion)	You are given a short visual description used for video retrieval, such as <*query*>. Your task is to rephrase the sentence while preserving **all** visual details, entities, and attributes. Do not remove or replace any element. Only change the phrasing or order of words. Provide twenty alternatives. Original: <*query*>
Qwen2.5-VL-7B (reranking)	How relevant is this image to the following caption: <*query*>? Answer with a single number 1-10.

query (Table 1) while strictly preserving all visual details, entities, and attributes. This process yields a set of augmented queries $Q = \{q, q_1, \ldots, q_k\}$.

We use five VLM families—CLIP [11], BLIP [12], BLIP-2 [13], SLIP [14], and BEiT3 [15]—along with their respective pretrained variants (listed in Table 2). All video shots in our dataset are processed using these models to extract their embeddings. Each augmented query in Q is then encoded using the same VLMs so that both queries and video shots lie in a shared embedding space. For every query–video pair, we compute cosine similarity, and aggregate the similarities within each VLM family using mean pooling to produce a single relevance score per family. Finally, we fuse the scores from all families using a weighted combination with fixed weights across VLM families. The final weights, i.e., $[10 : 5 : 5 : 1 : 20]$ for CLIP, BLIP, BLIP-2, SLIP, and BEiT3, respectively, were determined by internal experiments on the 2022 and 2023 queries. Following our 2024 approach [6], where we applied a similarity normalization procedure to enhance query–video shot matching, we utilize a normalization step to improve the comparability of similarity scores across queries. Specifically, for each query–video pair, we extend their final similarity score by comparing the same video shot against a set of background queries. This produces a vector consisting of the similarity between the target query and the shot, along with the similarities between the shot and all background queries. The resulting vector is then L2-normalized, ensuring that the normalized similarity of the target query accounts for how strongly the shot matches generic or unrelated background queries. After applying this normalization, the revised query-video shot similarity is the final similarity used for the retrieval stage, and using these similarity scores we generate a ranked list of the top-N most relevant video shots. As background queries, we consider the 2022, 2023, and 2024 AVS queries.

The second stage of our framework focuses on refining the initial top-N video shots returned in the retrieval stage for each input query. For this purpose, we employ the *Qwen2.5-VL-7B* model [16], an instruction-tuned MLLM capable of evaluating fine-grained semantic alignment between textual descriptions and visual content. Given a query and each of the top-N retrieved video shots, *Qwen2.5-VL-7B* jointly processes the text and visual frames and produces a contextual relevance score indicating how well each shot matches the intended semantics of the query. The prompt that was used is shown in Table 1. These scores are then used to revise the original query-video shot similarities, yielding a final ranking that more accurately reflects the query-specific visual relevance.

2.2 Submission

The V3C2 [17] dataset is utilized to evaluate the network's performance. Moreover, we examine the performance of our runs on the V3C2 datasets for the queries of years 2022-2023 as well as the performance using the 2024 ground truth in the same query set of 2025. The evaluation measure we use is the mean extended inferred average precision (MxinfAP).

This year, we submitted four runs for the AVS 2025 main task. The submitted runs are briefly described below:

- ITI_CERTH.25_run_1: Query expansion using $k = 20$ automatically generated rephrasings of the original query; Reranking applied to the retrieved video shots up to a depth of $N = 4,000$ candidates.
- ITI_CERTH.25_run_2: Similar to run_1, using $N = 2,000$ candidates in the reranking stage.

Table 2: The cross-modal network families and their pre-trained model variations utilized in our AVS 2025 participation.

Network Family	Pre-trained Model	Network Family	Pre-trained Model
BLIP	base_coco	BEiT3	base_coco (retrieval)
	base_flickr		base_flickr30k (retrieval)
	large_coco		large_coco (retrieval)
	large_flickr		large_flickr30k (retrieval)
BLIP-2	itm_coco	CLIP	RN101
	itm_pretrain		RN50
	itm_pretrain_vitL		RN50x4
SLIP	base_CC12M		RN50x16
	base_CC3M		RN50x64
	base		ViT-B/16
	large		ViT-B/32
	small		ViT-L/14

- ITI_CERTH.25_run_3: Similar to run_1, using $N = 1,000$ candidates in the reranking stage.
- ITI_CERTH.25_run_4: Base model; Similar to ITI_CERTH.25_run_1 without the reranking stage.

2.3 Experimental Results

Table 3: Mean Extended Inferred Average Precision (MXinfAP) for all submitted runs for the 2022, 2023, 2024 and 2025 fully automatic AVS tasks. Best results are **emphasized**.

Run id:	2022	2023	2024	2025
ITI_CERTH.25 run_1	**0.332**	**0.379**	**0.416**	**0.421**
ITI_CERTH.25 run_2	0.331	0.378	0.412	0.41
ITI_CERTH.25 run_3	0.312	0.357	0.381	0.369
ITI_CERTH.25 run_4	0.284	0.330	0.344	0.337

Table 3 presents the official results of our submissions for the main AVS 2025 task using the 2024 ground truth version and the updated 2025 one, as well as the results of our internal evaluation of AVS 2022 and 2023 queries. Overall, ITI_CERTH.25_run_1 consistently achieves the highest MXinfAP scores in all four years, indicating that combining query expansion with reranking over a 4,000-shot depth provides the most effective configuration. Reducing the reranking depth to 2,000 shots (ITI_CERTH.25_run_2) yields only a slight performance drop, indicating that the MLLM-based reranking remains effective even with fewer candidates. However, when the reranking depth is further reduced to 1,000 candidates (ITI_CERTH.25_run_3), performance decreases more noticeably across all years, suggesting that a larger reranking pool is beneficial for capturing relevant shots missed in early retrieval. The base system without reranking (ITI_CERTH.25_run_4) consistently produces the lowest scores, confirming the significant contribution of the MLLM reranking stage. The gains between ITI_CERTH.25_run_4 and the top-performing CERTH runs highlight the importance of the contextual reranking in improving retrieval performance. Across all runs, the results are stable over the four evaluation years, showing that the proposed two-stage foundation-model-based pipeline generalizes well across different query sets.

Figure 2 compares the performance of our submitted runs in the AVS 2025 main task against all submitted runs from the other participating teams, using the 2025 additionally-annotated ground truth data; Figure 3 shows the same comparison when using just the 2024 ground truth.

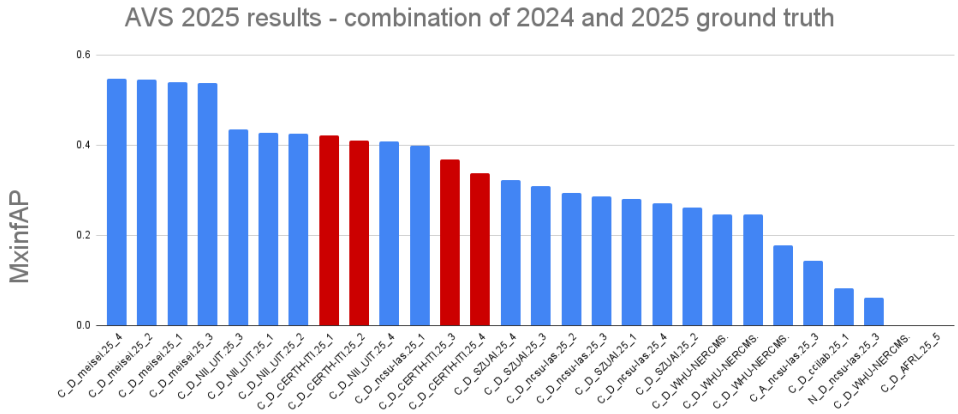

Figure 2: AVS 2025 ranking of all submitted runs using the combined ground truth of 2024 and 2025 evaluations, in MXinfAP terms. Red bars indicate our submitted runs.

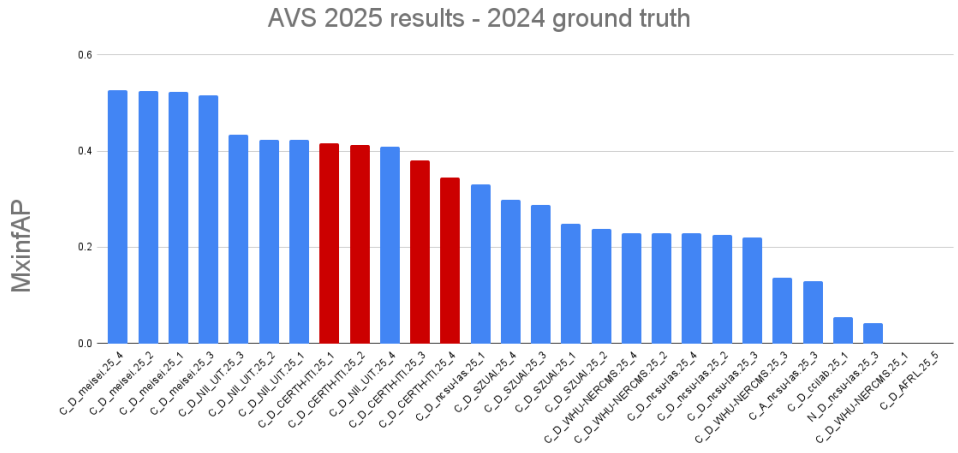

Figure 3: AVS 2025 ranking of all submitted runs using only the 2024 ground truth, in MXinfAP terms. Red bars indicate our submitted runs.

3 Video Question Answering

The TRECVID 2025 Video Question Answering (VQA) track consists of the following two tasks: a) Answer Generation (AG), where the objective is to generate a ranked list of up to 10 natural language responses to a video question prompt and b) Multiple Choice (MC), where the objective is to rank 4 candidate responses to a video question prompt. The evaluation dataset consists of 2,000 short videos that contain both audio and visual streams.

We address both tasks by employing a state-of-the-art audio-visual Multimodal LLM. We fine-tune the base model on the TRECVID-provided training dataset. During inference, we scale the multimodal understanding capabilities of the MLLM by aggregating multiple generations from different frame subsets of the video, as explained in detail in the next sections. The complete architecture is shown in Figure 4. We present the official evaluations of the runs and additional experiments to validate our methodology.

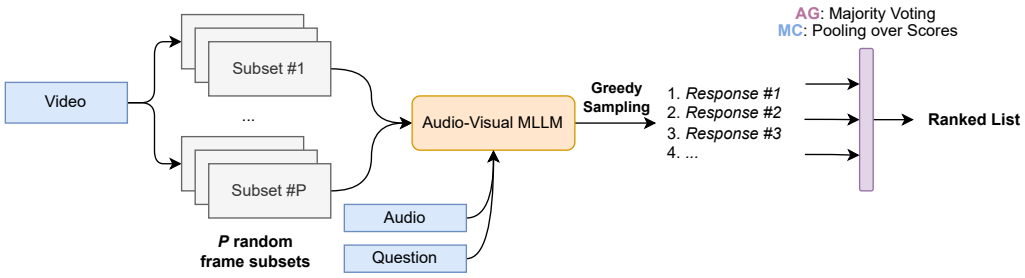

Figure 4: Video Question Answering architecture.

3.1 Approach

3.1.1 Base Model Selection and Fine-tuning

In order to address the requirements of the Video Question Answering tasks, we selected an omni-modal model, i.e. a model that is capable of processing both the visual and audio contents of the videos. Specifically, we used the Thinker component of Qwen2.5-Omni 7B [18] as our base model, while disabling the Talker component. We fine-tuned the Multimodal LLM using standard Supervised Fine-tuning (SFT) on the training dataset provided by the TRECVID organizers for this task[1]. This training dataset consists of 500 videos with audio and visual streams, along with associated question-answer pairs. In particular, for fine-tuning we used only the correct answers provided for each question and ignored the wrong answers. We fine-tuned the model on a single RTX 4090 24GB GPU using LoRA [19] for 5 epochs with batch size 32 and learning rate 3e-5. Due to memory and compute constraints, we opted to sample 10 frames per video during training and up to 1024 visual tokens per prompt. Regarding the LoRA adapter, we set $r = 16$ and $\alpha = 64$.

3.1.2 Frame Subset Ensembling

A well-known methodology for enhancing the capabilities of Large Language Models is self-consistency [20]. The main idea is to generate a diverse set of outputs and aggregate them, typically via majority voting, thus reducing the probability of error due to a single erroneous generation. In the video domain, Suo et al. [21] proposed a similar idea addressing the Long Video Understanding task. Specifically, they proposed sampling different subsets of frames bin-wise, generating different predictions for each subset, and aggregating the generations with a self-reward that is a linear combination of three scoring functions. Video Parallel Scaling (VPS) [22], which is contemporaneous with our work, structures this process as an inference-time parallel generation task and aggregates the output per-token with a weighted average. Finally, VidCtx [23] employs a similar frame-wise process, generating multiple predictions per video with an image MLLM that observes a single frame and relevant context, while aggregating the predictions across frames with max pooling.

Motivated by the above works, we adapt this paradigm to the open-ended audio-visual Video Question Answering task. In contrast to Suo et al. [21], we employ a simple majority voting aggregation layer, without introducing external supervision from additional expensive models. Critically, we generate one LLM completion per frame subset using greedy sampling. Greedy sampling significantly reduces the exploration factor during auto-regressive generation, which enables us to utilize majority voting with open-ended free-form responses. Therefore, the response diversity is exclusively due to the sampling of different frames per generation. Furthermore, we do not use bin-wise frame sampling, considering the short duration of the videos.

We sample $P = 10$ frame subsets randomly from the entire video, considering all frames. Each subset can have up to $N = 2fps \times Tsec$ frames, where T is the duration of the video in seconds. However, due to the sampling process, the FPS of each subset is not constant throughout the entire duration. Furthermore, we combine the audio embeddings of the entire video with the sampled frames as-is. During inference, we set the maximum number of visual tokens to $16,384$ for each generation.

[1]Training dataset can be found here: `https://www-nlpir.nist.gov/projects/tv2025/vqa.html`

Finally, we rank the generated responses based on the number of occurrences, i.e. the number of votes, of each response.

For the VQA Multiple-Choice (MC) task, we follow a similar approach as described above. Specifically, we instruct the Multimodal LLM to generate the letter of the predicted answer (i.e., A, B, C, D). Following [23] and [21], we utilize a logit pooling layer instead of majority voting. This layer aggregates the log-probabilities (logits) assigned by the Multimodal LLM to the letters of the candidate choices across the responses. Specifically, we use mean pooling and we rank the responses according to the pooled probabilities. In addition, we experimented with ranking the candidate responses by estimating the confidence assigned by the model to each one, via the mean log probability of the response tokens. However, in our experimental setup, this method of assigning scores to each candidate response significantly underperformed the direct generation of the letter of the predicted answer, as shown in the next section.

3.2 Submission - Answer Generation

For the VQA Answer Generation task, we submitted the following 4 runs:

- certh.vqa.ag.run_1: (Main run) Aggregation of $P = 10$ predictions per question from different frame subsets of the video, generated with the fine-tuned model via greedy sampling.
- certh.vqa.ag.run_2: $P = 1$ generation per question with the fine-tuned model.
- certh.vqa.ag.run_3: Same as Run 1, but with multinomial sampling instead of greedy sampling.
- certh.vqa.ag.run_4: Same as Run 1, removing any quote characters from the final responses as a post-processing step. Run 4 was submitted as a check to determine whether the evaluation platform parsed the submitted CSV files as intended.

3.3 Submission - Multiple Choice

For the VQA Multiple Choice task, we submitted the following 4 runs:

- certh.vqa.mc.run_1: Base model without fine-tuning. Aggregation of $P = 10$ predictions with mean pooling of the candidate logits.
- certh.vqa.mc.run_2: Fine-tuned model. Aggregation of $P = 10$ predictions with mean pooling of the candidate logits.
- certh.vqa.mc.run_3: Base model without fine-tuning. Aggregation of $P = 10$ predictions, where the answer is chosen by confidence estimation.
- certh.vqa.mc.run_4: Fine-tuned model. Aggregation of $P = 10$ predictions, where the answer is chosen by confidence estimation.

3.4 Experimental Results

3.4.1 Results for the Answer Generation Task

We present the official results of our submitted runs in Table 4. The runs are evaluated based on the following 3 metrics, averaged across the responses: a) METEOR score, b) BERT semantic similarity score and c) Semantic Textual Similarity (STS) score.

Firstly, we observe that our main run (Run 1) significantly outperforms Run 3, which uses multinomial sampling instead of greedy sampling, due to the reduced randomness in the generated responses. Secondly, Run 1 and Run 4 have identical performance, since the only difference between the two runs is the removal of all quote characters from the final responses, as explained in the previous section. Therefore, we de-emphasize the results of Run 4 in the table. Third, the difference between our main run (Run 1) and the single-generation baseline (Run 2) is more subtle. However, as we explain in the next section, the two runs are not directly comparable, considering the different number of responses submitted per question between the two.

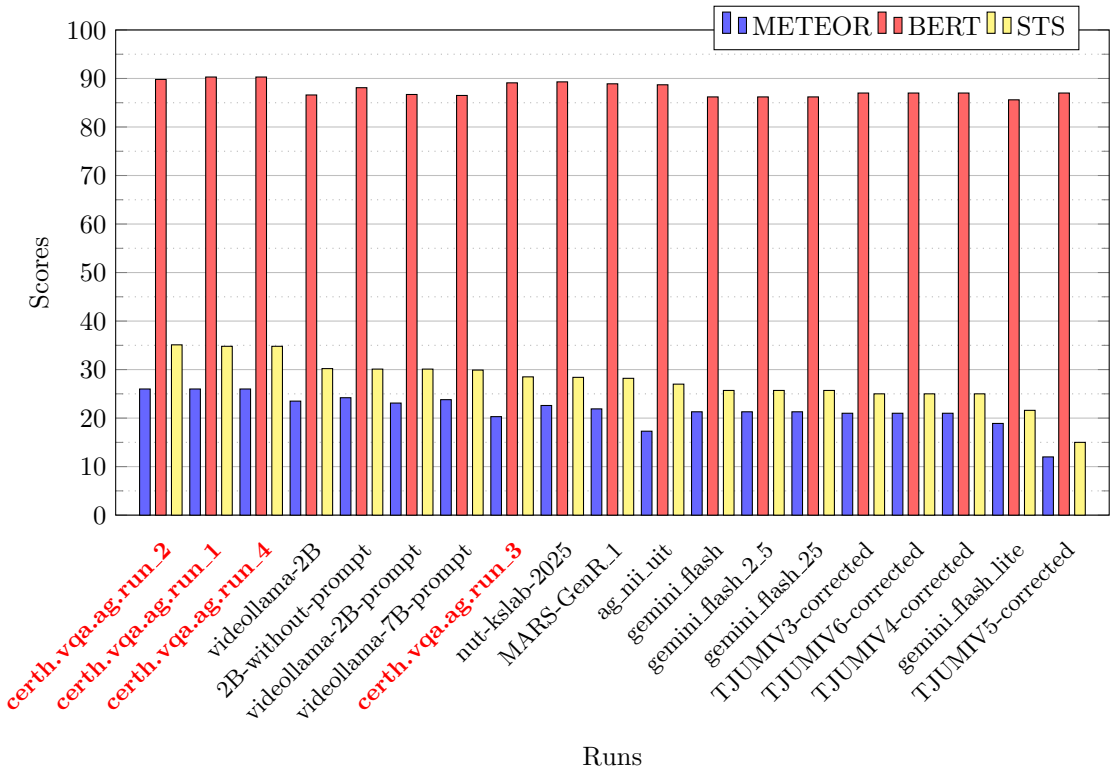

Figure 5: VQA Answer Generation (AG) ranking of all submitted runs in TRECVID 2025, sorted by STS score. Run IDs in red on the horizontal axis indicate our submitted runs.

Figure 5 shows the performance of our submitted runs in the VQA AG track compared to all submitted runs from the other participating teams, sorted by STS score. We observe that we achieve the highest scores among all participants, across all 3 evaluation metrics.

3.4.2 Discussion on Answer Generation Task and Additional Evaluation

We observe that the aggregation of the metrics (METEOR, BERTScore, STS) across the ranked responses penalizes the methods that return more responses per question. Even if the response with the highest rank is correct, the presence of additional responses has a significant negative impact on the final score. Therefore, for a fair comparison, we present an additional evaluation of the runs, where we keep the highest ranked answer and discard the rest (i.e., employing top-1 selection). We present the results of this evaluation in Table 5. This process ensures that all runs have exactly one response. We calculate the new metrics on the basis of the provided per-response evaluation CSV

Table 4: Official results for the submitted runs in the VQA Answer Generation (AG) task. Best results are **emphasized**. The results of Run 4 are de-emphasized because the run is the same as Run 1, with all quote characters removed.

Run id:	METEOR	BERTScore	STS
certh.vqa.ag.run_1	**0.260***	**0.903***	0.348
certh.vqa.ag.run_2	**0.260***	0.898	**0.351***
certh.vqa.ag.run_3	0.203	0.891	0.285
certh.vqa.ag.run_4	0.260	0.903	0.348

* We denote with an asterisk the runs that achieved the highest score among all participants in TRECVID 2025 for each metric.

Table 5: Results with top-1 selection for the VQA Answer Generation (AG) task. Best results are **emphasized**.

Run id:	METEOR	BERTScore	STS
certh.vqa.ag.run_1 + Top-1	**0.265**	**0.905**	**0.356**
certh.vqa.ag.run_2 + Top-1	0.260	0.898	0.351
certh.vqa.ag.run_3 + Top-1	0.215	0.897	0.304

files. We observe that under this setup, our main run (Run 1) significantly outperforms the baseline approach (Run 2) across all metrics, validating our methodology. In addition, we see that multinomial sampling (Run 3) continues to have a significant negative impact on performance, as explained above.

3.4.3 Results for the Multiple Choice Task

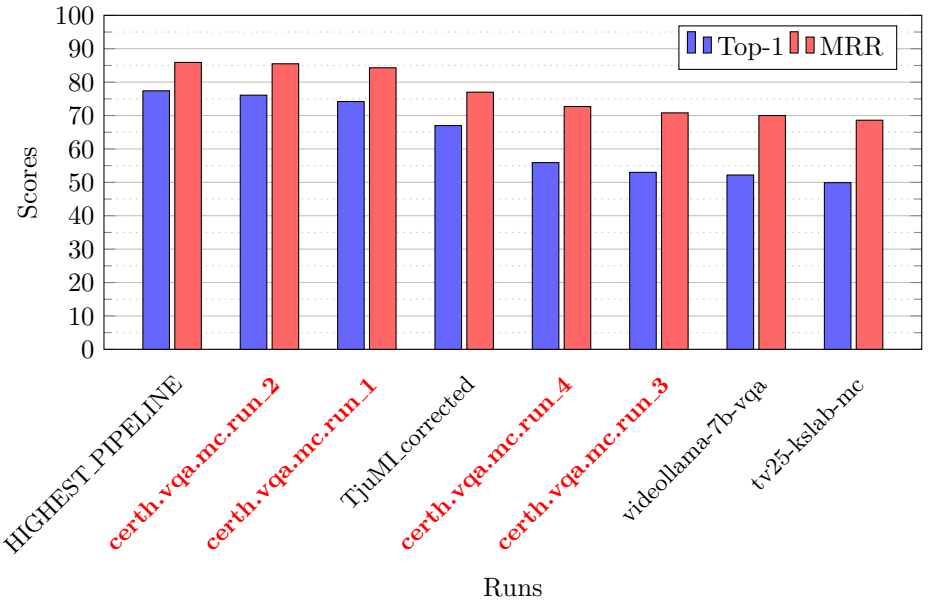

Figure 6: VQA Multiple Choice (MC) track ranking of all submitted runs in TRECVID 25 sorted by top-1 accuracy. Run IDs in red on the horizontal axis indicate our submitted runs.

In Table 6, we present the official results of the submitted runs in the Video Question Answering Multiple Choice (MC) task. The metrics that were used to evaluate the runs are a) the top-1 accuracy and b) the mean reciprocal rank. Firstly, we observe that both for direct generation and confidence estimation, our fine-tuned model outperforms the base Qwen2.5-Omni model, even though it was trained on open-ended question-answer pairs. Secondly, we observe that the direct generation method significantly outperforms the confidence estimation method.

Figure 6 shows the performance of our submitted runs in the VQA MC track compared to all submitted runs from the other participating teams, sorted by top-1 accuracy. We observe that our approach achieves the second highest score across all participants.

To validate the performance of the scaling process, we performed additional ablation experiments, which are presented in Table 7. We evaluate the new runs based on the provided ground truth responses from the organizers. Based on these experiments, we conclude that the inference-time technique provides a clear performance gain in both setups.

Note about the evaluation. We observed that a small number of multiple-choice questions and candidate answers in the provided evaluation dataset contained incorrectly escaped quotes (i.e., a single quote character instead of the expected double quote). We manually escaped the quotes for

Table 6: Official results for the VQA Multiple Choice (MC) task. Best results are **emphasized**. SFT refers to Supervised Fine-tuning.

Run id:	Variant	Top-1 (%)	MRR
certh.vqa.mc.run_1	zero-shot	74.2	0.843
certh.vqa.mc.run_2	SFT	**76.1**	**0.855**
certh.vqa.mc.run_3	zero-shot	53.0	0.708
certh.vqa.mc.run_4	SFT	55.9	0.727

Table 7: Ablation study for the VQA Multiple Choice (MC) task. Best results are **emphasized**.

Method	Top-1 (%)	MRR
Base model	73.2	0.838
+ multiple generations	74.2	0.843
Fine-tuned model	74.4	0.845
+ multiple generations	**76.1**	**0.855**

those entries prior to submitting the official results. However, according to the provided diagnostic CSV files, these responses were all marked as incorrect by the automatic evaluation system due to incorrect parsing. This issue affects 0.7% of the questions. In order to present a fair comparison of the ablation experiments with the official runs that we submitted, we, too, mark those answers as incorrect in the ablation study.

4 Conclusions

In conclusion, in the AVS track, we presented a two-stage AVS framework that leverages foundation models for both retrieval and reranking. We improve the performance of our pipeline through MLLM-based reranking; this revises the initial retrieval results by improving the semantic alignment between the query and the top-ranked shots, consistently producing more accurate and contextually relevant final rankings.

Regarding the VQA track, through the combination of fine-tuning and inference-time scaling with the aggregation of multiple predictions per question, generated from different frame subsets, we achieve the highest score across the employed 3 metrics in the Answer Generation task, and the second-highest score in the Multiple Choice task. Furthermore, we show through additional experiments the effectiveness of each component of our framework and we present additional evaluations of the submitted runs, for a more direct comparison.

5 Acknowledgements

This work was supported by the EU's Horizon Europe programme under grant agreements 101070190 AI4Trust and 101070109 TransMIXR.

References

[1] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[2] Ian Soboroff and George Awad. Overview of TREC 2025. In *Proceedings of TRECVID 2025*. NIST, USA, 2025.

[3] George Awad. TREC 2025 Ad-hoc Video Search (AVS) Track Overview. In *Proceedings of TRECVID 2025*. NIST, USA, 2025.

[4] George Awad, Sanjay Purushotham, and Afzal Godil. Video Question Answering (VQA) 2025 Track. In *Proceedings of TRECVID 2025*. NIST, USA, 2025.

[5] Damianos Galanopoulos and Vasileios Mezaris. ITI-CERTH participation in AVS Task of TRECVID 2023. In *TRECVID 2023 Workshop, Gaithersburg, MD, USA*, 2023.

[6] Konstantinos Gkountakos, Damianos Galanopoulos, Antonios Leventakis, Georgios Tsionkis, Klearchos Stavrothanasopoulos, Konstantinos Ioannidis, Stefanos Vrochidis, Vasileios Mezaris, and Ioannis Kompatsiaris. ITI-CERTH participation in AVS Task of TRECVID 2024. In *TRECVID 2024 Workshop, Gaithersburg, MD, USA*, 2024.

[7] An Yang, Baosong Yang, Beichen Zhang, and et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[8] Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[9] Meta. Llama-3.2-1b-instruct, 2024. Accessed: 2025-10-05.

[10] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[11] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, , et al. Learning transferable visual models from natural language supervision. In *Proc. of the 38th Int. Conf. on Machine Learning (ICML)*, 2021.

[12] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[13] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[14] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *European conference on computer vision*, pages 529–544. Springer, 2022.

[15] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: BEiT pretraining for vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[16] Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2.5-VL technical report. Technical report, arXiv, 2025. Technical Report, Qwen2.5-VL series.

[17] L. Rossetto, H. Schuldt, G. Awad, and A. A. Butt. V3C–a research video collection. In *Proc. of MMM 2019*, pages 349–360. Springer, 2019.

[18] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, et al. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025.

[19] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[20] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[21] Yucheng Suo, Fan Ma, Linchao Zhu, Tianyi Wang, Fengyun Rao, and Yi Yang. From trial to triumph: Advancing long video understanding via visual context sample scaling and self-reward alignment. *arXiv preprint arXiv:2503.20472*, 2025.

[22] Hyungjin Chung, Hyelin Nam, Jiyeon Kim, Hyojun Go, Byeongjun Park, Junho Kim, Joonseok Lee, Seongsu Ha, and Byung-Hoon Kim. Video Parallel Scaling: Aggregating Diverse Frame Subsets for VideoLLMs. *arXiv preprint arXiv:2509.08016*, 2025.

[23] Andreas Goulas, Vasileios Mezaris, and Ioannis Patras. VidCtx: Context-aware Video Question Answering with Image Models. In *IEEE Int. Conf. on Multimedia and Expo (ICME 2025)*. IEEE, 2025. https://doi.org/10.1109/ICME59968.2025.11210080.