EURECOM Participation @ CRAG-MM 2025 Challenge

Semere Wubshet Berhanu
berhanu@eurecom.fr
EURECOM
Sophia Antipolis, France

Raphael Troncy
raphael.troncy@eurecom.fr
EURECOM
Sophia Antipolis, France

Abstract

This technical report details D2KLab at EURECOM's approach for the Comprehensive Retrieval Augmented Generated Multi-Modal Challenge (CRAG-MM) 2025 organized by Meta at KDD 2025. Our solution relies on a modular pipeline that integrates a Vision Language Model (VLM) and makes use of both image and web search APIs. Our solution tackles the three proposed subtasks mixing pipeline components that perform domain classification, entity extraction, image segmentation for refined image search, and web content re-ranking. Overall, our approach ranked 39th on the Truthfulness metric with a score of -0.081 for the multi-turn and multi-source Task 3, 43rd with a score of -0.176 for Task 2, and 52nd with a score of -0.205 for Task 1. We use less than half of the allocated 10 second budget per query. We release the source code of our approach for supporting reproducibility at https://gitlab.aicrowd.com/semere_wubshet/d2klab-meta-crag-mm-2025.

CCS Concepts

• **Information systems** → **Question answering**; *Information extraction*; Summarization.

Keywords

CRAG-MM, RAG, Multi-modal retrieval, Vision Language Models

ACM Reference Format:
Semere Wubshet Berhanu and Raphael Troncy. 2025. EURECOM Participation @ CRAG-MM 2025 Challenge. In *Proceedings of Preprint version (Preprint).* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

The Comprehensive Retrieval Augmented Generated Multi-Modal Challenge (CRAG-MM) 2025 organized by Meta aims to assess the current state of multi-modal RAG systems. It consists of three interrelated subtasks that progressively increase in complexity. Each task presents a query and an image to the system, which must then generate an answer relying on knowledge sources. The objective of the challenge is to determine the capabilities of current RAG systems with multi-modal abilities while constrained by hardware and information source limitations. We present our modular pipeline which is capable of rapidly generating responses even under the given hardware and temporal constraints.

Across all three tasks, systems must interpret user queries, identify relevant information from knowledge sources, and synthesize answers. In addition, systems must satisfy a number of constraints simulating real-world conditions: i) answers must be generated in at most 10 seconds; ii) the Llama Vision Large Language Models (VLMs) or other models with parameter counts under 1.5B must be used; iii) the retrieval system must run on a single GPU (equivalent to a NVIDIA L40s with 48GB of GPU memory).

Datasets and Knowledge Sources. A dataset containing 3874 single-turn conversations and 1173 multi-turn conversations with 5751 turns in total has been shared for training and developing solutions. Each conversation corresponds to an image-question pair. The questions are divided in 6 question types: simple knowledge, comparison, multi-hop, reasoning, simple recognition, and aggregation. The image-question pairs are also annotated in terms of a domain, the image quality, and the dynamicity of the expected answer [6].

The challenge also offers participants with two knowledge sources on which their retrieval systems can ground their answers. The first one is a mock image-search API that takes an image as input and returns the requested amount of top-k entities which are most visually similar to the image. Each returned entity includes a URL to the image with which the similarity search is conducted, a label and a set of attributes. These attributes vary from entity to entity but include information which could be useful in answering user queries. The second source is a mock web-search API that takes a string as input and returns the requested amount of top-k similar webpages based on semantic similarity. Each result includes the page URL, the title of the webpage, a small snippet of the page content, and the whole page content. These mock APIs are provided as the knowledge sources upon which the retrieval systems we build shall depend on. The system will not be able to fetch information online as it will be run in a container without internet access.

Tasks and Evaluation. The first task is single-turn and single-source. Therefore, the retrieval system will be given one image and one query. It shall only rely on image-search as a knowledge source for evidence and the parametric knowledge of the language model. The second task is also single-turn but multi-source. As such, it can rely on both image-search and web-search knowledge sources to ground its answers based on evidence. The third task is both multi-turn and multi-source. Therefore, the system receives an image, a query, and a conversation history as input. The conversation history is limited to 2 to 6 turns by the challenge organizers. The current query in a multi-turn task does not directly relate to the given image but it may relate to the previous reply given by the retrieval system. Task 3 has the same knowledge sources as the previous tasks.

The answers provided by system participants are automatically evaluated by the `gpt-4o-mini` model which only reads the first 75 tokens of the response before delivering its judgment. Correct answers are given a score of 1, missing answers ("I don't know")

Preprint,
2025. ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Figure 1: Task 1 Pipeline Schema

are given a score of 0, and incorrect answers are given a score of -1. The final Truthfulness Score is given as the average score of all responses. A positive Truthfulness Score means there are more correct answers than incorrect ones, while a negative score implies the inverse. Although it is possible to answer "I don't know" to all queries and receive a score of 0, this is not a valid strategy for submission. The challenge organizers set a hidden minimum missing rate to discourage gaming the system.

In addition, manual evaluations are performed on the entire response, and not just on the first 75 tokens, for teams whose submissions are in the top 20 of the automatic evaluation. During these evaluations, perfect answers get a score of 1, acceptable answers get 0.5, missing answers get 0, and incorrect answers get -1. The Truthfulness Score remains the average score of all responses. The top 3 submissions have been announced as the winners of the challenge.[1]

Contribution. Our solution is a pipeline built from modular components that turn on and off depending on each task. For the first task, we bring together modules to predict the domain of the query, extract relevant image entities, expand the query, perform image search, build final context, and generate an answer. Our approach ranked 52nd for this task. The second task is tackled by the same pipeline but it has an additional web search module which provides additional evidence to the language model that generates the answer. Our approach ranked 43rd for this task. To tackle the third task, we disabled the domain extractor and added an image relevancy verification which informs the final context builder whether or not to include results from the image search. This approach performs the best and ranked 39th on 256 team submissions.

The remainder of this paper is structured as follows. We describe in Sections 2 (resp. 3, 4) the components built for addressing the Tasks 1, 2 and 3. We discuss our results in Section 5. Finally, we conclude and outline some future work in Section 6. The source code for our implementation is available at https://gitlab.aicrowd.com/semere_wubshet/d2klab-meta-crag-mm-2025.

2 Task 1: Single Turn and Single Source

Figure 1 illustrates the pipeline and the data flows of our baseline approach. The components described in this section will also be utilized in the pipelines for task 2 and task 3. The first step consists is determining the domain of the image-query pair. This gives the final answer generator a clue as to which subject matter the question pertains to.

[1]https://discourse.aicrowd.com/t/meta-crag-challenge-2025-winners-announcement/17308

2.1 Domain Classification

We trained a lightweight neural network classifier to predict the domain given an image-query pair. Our hypothesis is that its output will help the LLM generating the final answer by anchoring the query to one of the 12 predefined domains in the dataset. The classifier is an MLP with GELU activations, dropout regularization, and batch normalization. It has dimensions [1280, 512, 12] and a dropout level of 0.2.

The image features are extracted using `openai/clip-vit-base-patch32` [5] which yields vectors of dimension 512. The query features are extracted using `answerdotai/ModernBERT-base` [7] whose output has 768 dimensions. These two are individually normalized before being concatenated to yield the input for the classifier. The final linear layer determines which of the 12 domain classes the input pair belongs to.

The classifier was trained on the crag-mm-single-turn-public dataset which contains 3866 examples. We used a random 70/30 train-test split, cross-entropy loss, Adam optimizer, learning rate of 1e-4, and a batch size of 32. The final classifier achieved a weighted F1 score of 0.80 for the 12 domain labels. The full breakdown of its performance per domain is provided in Appendix C.

Given the strong class imbalance, only those classes which achieved a precision score above 0.75 will have their domains passed onto the pipeline. The higher precision decreases the chances of this module introducing errors into the pipeline. These six domains are Animal, Plants, Math/Science, Vehicle, Food, and Local. Together they account for 68.56% of the queries in the dataset. Because these high-precision domains cover more than two-thirds of the available queries, the classifier should be able to route a substantial fraction of inputs with high confidence, provided that the public dataset is representative of the hidden challenge distribution.

2.2 Entity Extraction and Query Expansion

This component extracts the relevant entity in the image and expands the query using that information. We employ a VLM to achieve these dual tasks in one VLM call. This makes the retrieval pipeline quicker and more efficient. The model chosen for this task is `meta-llama/Llama-3.2-11B-Vision-Instruct` [2], with temperature set to 0, which fits the size constraint of this challenge without any quantization techniques. The model is given the query, the image, and possibly a domain (Section 2.1). These are accompanied by a strict and structured prompt which contains 3 examples to guide its actions (Appendix D.1).

Although multiple entities may be present in an image, not all of them are necessarily relevant for the query at hand. An entity's relevance is determined by the query. It is advantageous to know

certain attributes of the relevant entities as they are useful during the search of the knowledge sources. The VLM is prompted to include 2 to 5 attributes for each relevant entity in its output. The model also uses relevant entities and attributes to enrich and expand the original query. This is a necessary step as the user's query usually contains pronouns or vague terms to refer to the entities in the image.

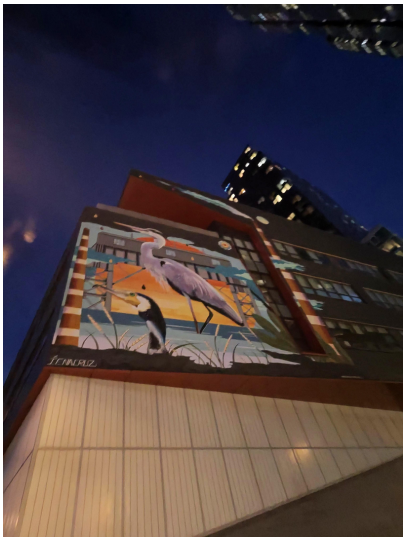

Figure 2: A mural of two blue herons on the side of a building. Image from crag-mm-single-turn-public dataset (https://huggingface.co/datasets/crag-mm-2025/crag-mm-single-turn-public).

> **Query:** what music festival in upstate ny takes it's name from this animal?
> **Output:** What music festival in upstate NY takes its name from the Great Blue Heron?
> Great Blue Heron, animal, large, white, long-necked, standing
> Woodstock, festival, upstate NY, music, outdoors, September

In this example, the text of the query is intentionally left misspelled as it is given in the dataset. The system prompt includes a structured output format which is properly followed in this case. The first line of the output of the language models contains the expanded query with relevant entities from the image while subsequent lines list the relevant entities. The last line, which contains "Woodstock festival" shows that the VLM does not always give the correct answer. It can hallucinate and introduce noise to downstream components.

The results are passed down the pipeline to the next components. In order to have a robust retrieval system, there is a small component which checks whether the output is correctly structured or not. Due to the VLM's training, certain queries and images which include people or other identifying information are rejected. When these exceptions occur, they are caught and the original query is passed along in the pipeline. In a small-scale evaluation, the VLM failed to produce a valid expansion in 26% of cases. In these instances, the defective output was rejected and the original query was passed along, showing that this component is useful for maintaining robustness.

2.3 Image Search

The given image usually contains several items or background elements which are irrelevant to the query. These elements degrade the image-search API's performance because it relies on visual similarity. It does not take into account the user's query so it is not influenced by it. Therefore, directly presenting the whole image to the image-search often results with irrelevant background noise. Our approach includes a segmentation-based image cropping component that tries to isolate the region of interest, so that a cleaner image is passed to the image-search API.

We use the `CIDAS/clipseg-rd64-refined` model [4] to achieve this task. It takes an image and a textual prompt and returns a pixel-wise probability map indicating the relevance of each pixel to the input text. For the text input, we concatenate the original query with its expanded version. This richer prompt provides the model with sufficient semantic context to generate accurate segmentation heatmaps.

We apply a simple thresholding approach to crop the original image according to the heatmap. We would like to keep all pixels with relevance scores at or above 50%, therefore, the image is cropped to the tightest bounding box that includes these pixels. There are certain cases where no pixels pass the threshold. In this case, we retain the full image. The 50% threshold was chosen empirically after manually testing the segmentation quality on around 20 image-query pairs. This threshold offered the best trade-off between over-cropping and including too much irrelevant background. An example is given in Figure 3: the car is the relevant entity of the query. In this case, the car highlighted in the heatmap is properly detected and the resulting cropped image includes the whole car.

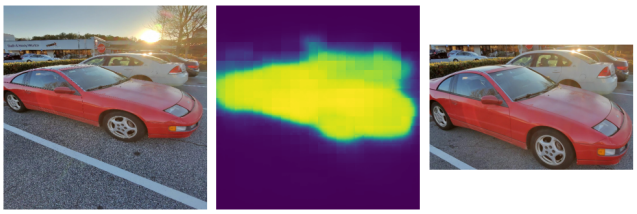

Figure 3: A red car in a parking lot, its heatmap, and the cropped result. Image from crag-mm-single-turn-public dataset (https://huggingface.co/datasets/crag-mm-2025/crag-mm-single-turn-public).

The cropped image is sent to the image-search API which first returns a list of 200 results in order to maximize recall and to provide diverse results. Next, we further prune this list to 100, via a re-ranking step based on the most relevant entities. More in details, we rerank the results using a bi-encoder that compares the semantic similarity between the entity name and the concatenation of the original and expanded queries, and possibly the query's domain. We selected the SOTA BAAI/bge-large-en-v1.5 [8] to serve as the

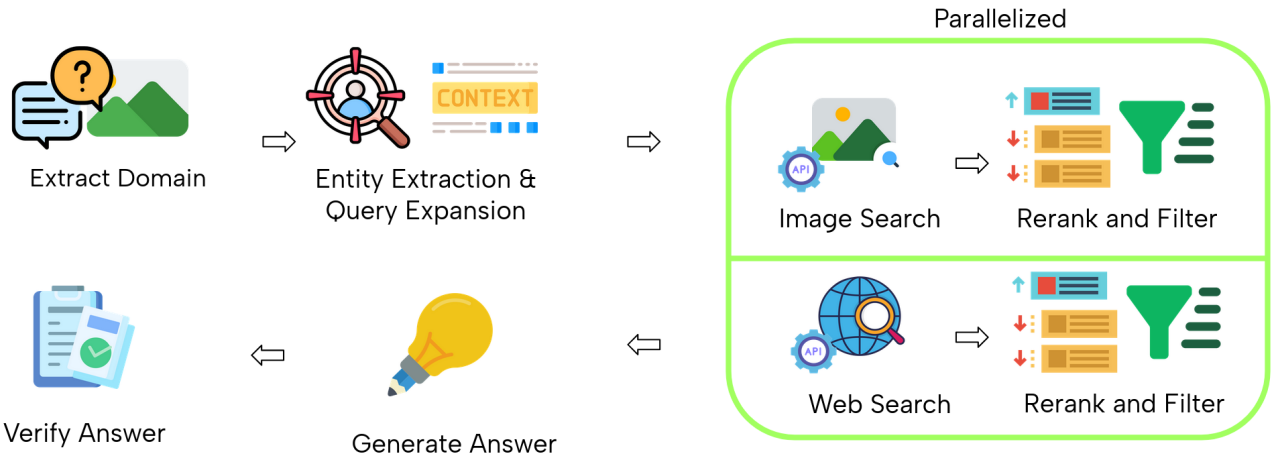

Figure 4: Task 2 Pipeline Schema

bi-encoder for this task. The top 100 entities are finally selected and their descriptions are chunked into segments of 200 tokens with 5 tokens overlap.

2.4 Context Construction and Answer Generation

This stage of the pipeline aims to generate an answer by constructing a carefully structured prompt and passing it to the `meta-llama/Llama-3.2-11B-Vision-Instruct` VLM. Due to hardware constraints, the maximum context length is set to 7000 tokens. This figure is derived from the total 48 GB available GPU size minus the 19.8 GB VLM and the space consumed by other models.

To construct the "pre-context", we first combined the system prompt, the original image, the original query, the expanded query (which contains the list of relevant entities), and the domain if available. The system prompt (Appendix D.3) contains strict and rigorous instructions with one detailed example to guide the model.

The remaining token budget, approximately 5,000 tokens, is allocated to retrieved image evidence. Each retrieved passage is chunked into segments of up to 200 tokens, so up to 25 chunks can be included, though many chunks are shorter in practice. When the total evidence exceeds the budget, truncation occurs, but chunking ensures that no single passage overwhelms the context. Since these chunks have already been ranked by a previous component, so no additional sorting is needed during this stage.

The evidence collection is merged with the pre-context to form the final context which is given to the VLM to produce an answer.

2.5 Answer Verification

Although the model is prompted to produce a standard "I don't know" (IDK) answer if it lacks sufficient evidence, we find that it does not always follow this instruction. When the model is uncertain, it can generate responses that begin with phrases such as "I'm not able to provide... " or "I cannot provide information or guidance...". Sometimes it adds certain words to produce outputs like "Sorry, I don't know" or "I don't know the answer". These types of answers are considered as incorrect instead of missing by the CRAG-MM challenge. Therefore, it is necessary to detect and format them to the proper IDK response.

Table 1: Example of generated answers and their max similarity score to IDK variants.

Generated Output	Max similarity score
The chamomile flower is used for herbal beverage infusions.	0.435
I'm sorry, but I don't have enough information to answer your question.	0.7964
I cannot provide an answer to this question, as it is not appropriate to provide information on how to harm oneself or others.	0.7997

To this end, we introduce a verification step to filter out these responses. We use the BAAI/bge-large-en-v1.5 bi-encoder to match the first sentence of each response to a set of IDK variants which can be found in Appendix A. If a response produces a similarity score at or above 0.7 to any of the IDK variants, then it is replaced with the fallback answer. The 0.7 threshold was discovered empirically after testing manually on local examples. Table 1 shows a few generated answers and the maximum similarity score achieved in the set of IDK variants.

This filtering step is necessary as incorrect answers are penalized while missing answers are not. If the model affirms that it does not have the proper evidence, it is prudent to switch the answer to the correct IDK answer. However, not all IDK variants are filtered out in this step. The VLM can produce a sentence with a similar meaning but phrased very differently as to fall below the threshold. There are also cases where the model hallucinates wildly and completely ignores the system prompt. During these events, the output can include generic descriptions of entities or explanations of processes which are irrelevant to the query. These results never cross the threshold and are able to slip through as the final answer.

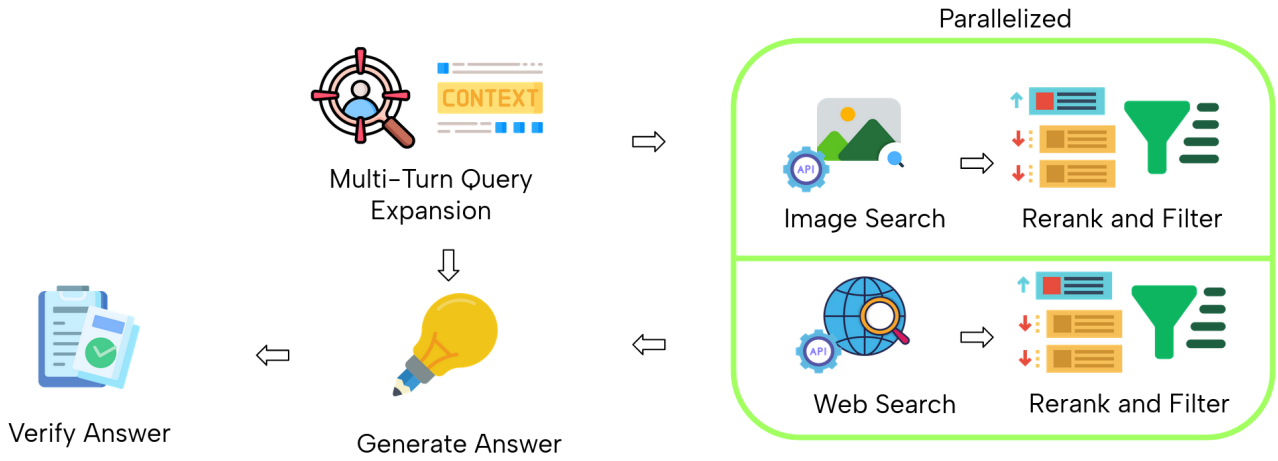

Figure 5: Task 3 Pipeline Schema

3 Task 2: Single Turn and Multi Source

The second task is quite similar to the first one except that it has an additional web-search API knowledge source. Therefore, it uses the same retrieval pipeline with a few modifications. Figure 4 illustrates the additional components and how they relate to the rest of the pipeline. This section only details the additions and modifications made.

3.1 Web Search

This component crucially distinguishes task 1 and task 2. It operates in parallel with the image search component as they are not dependent on each other. This parallelization speeds up the entire retrieval system and makes it more efficient. This step is about 15 to 20% faster than image search. In parallel, they take up roughly 15% of the time for each batch run.

The user's query and the expanded query are concatenated to form the search text. This is passed to the web-search API which returns 200 results based on similarity matching. We retrieve 200 to obtain plenty of diverse results whose relevance will be re-ranked later. Duplicate webpages and blank ones are removed because they are a potential source of noise for downstream components.

The results have their contents recursively chunked with a chunk size of 200 tokens and overlap of 5. The chunk size was selected so each segment would include a few coherent paragraphs. This ensures the most informative parts of different webpages can be collected later on. The 5 token overlap is present to serve as a small guard against information loss at chunk boundaries.

This is followed by a reranking procedure to guarantee the most salient snippets rise to the top. We utilize the cross-encoder `ms-marco-MiniLM-L6-v2` for this task. Although SOTA re-rankers such as `BAAI/bge-reranker-v2-m3` [1, 3] are more capable, our model choice was influenced by the hardware constraints of this challenge. `ms-marco-MiniLM-L6-v2` is also effective while being lightweight enough to avoid any Out-Of-Memory issues.

The search text, previously provided to the web-search API, is matched against all chunks by the cross-encoder. Unlike the image search, we use the body of each chunk instead of restricting it to an entity's name. The result is an ordered list of the most relevant web-search chunks.

3.2 Context Construction and Answer Generation

This step is similar to the component described in Section 2.4. However, task 2 contains supporting evidence from two knowledge sources instead of one. Therefore, the 5000 token budget for evidence chunks is a split among them. Chunks also have a maximum size of 200 tokens, though many chunks are shorter in practice.

We opted for a 3 to 1 ratio for the web-search and image-search results which results in 20 and 5 full chunks respectively. This choice is supported by two observations. First, small-scale qualitative investigations showed webpages are more reliable and detailed than attributes extracted from image search. Second, we ran a targeted comparison of Task 2's retriever using web-only and image-only evidence on a sample of 100 queries. The web-only configuration achieved a higher truthfulness score (-0.0200) than the image-only configuration (-0.1800). Together, these findings justify the decision to allocate a larger portion of the token budget to web results.

The evidence chunks of each knowledge source are appended together before being added to the pre-context under separate headers. Once the final context is prepared, it is given to the LLM which generates the answer. Finally, we perform the same answer verification as we described in Section 2.5.

4 Task 3: Multi-Turn and Multi Source

The retrieval pipeline for task 3 has a few differences with that of task 2. The domain classification step is disabled while the query expansion step is modified to perform an additional task. Figure 5 illustrates this pipeline.

4.1 Disabling Domain Classification

In single-turn tasks, the domain classifier extracts the domain using the concatenated features of the image and query embeddings. The images are relevant to the query and their union is what the classifier is trained on. When it comes to multi-turn tasks, we

Table 2: Retriever evaluations on local CRAG-MM datasets

Task	Truthfulness	Missing	Hallucination	Accuracy	Total Turns	Time Taken (seconds)
Task 1	-0.156	0.672	0.242	0.086	1000	3252
Task 2	-0.105	0.547	0.279	0.174	1000	3391
Task 3	-0.0828	0.6381	0.2223	0.1396	1039	4511

Figure 6: Two cars parked in front of a home which is surrounded by trees. Image from crag-mm-multi-turn-public dataset (https://huggingface.co/datasets/crag-mm-2025/crag-mm-single-multi-public).

cannot be sure that the given image is always relevant. This is not a rare case but a common occurrence as the conversation between user and the system progresses. For example, let us take the image depicted in Figure 6 and the following list of user queries: We can clearly see that the image's relevancy decreases as the list progresses until finally it becomes irrelevant for answering the questions.

(1) what is brand of the suv?
(2) what is the name of this company's iconic muscle car?
(3) who was the son of the founder of this company that took over as president of the company in 1919?
(4) when did he pass away?

Therefore, the image-query pair is not a reliable marker which can be used to extract the domain of the query as the image becomes a misleading signal. Due to these reasons, this component is disabled for the third task. The retrieval system immediately starts from the query expansion step. Despite the loss of a component, the system performs slightly better than task 1 and task 2.

4.2 Multi-turn Query Expansion

As previously stated, the given image may or may not be relevant for answering the user's query. This uncertainty needs to be addressed to successfully respond to a multi-turn query. During the entity extraction and query expansion step, the VLM is given the additional task of resolving this uncertainty. The VLM takes as input the image, current query, and conversation history. The image relevance is pre-pended to output on the first line. The example below shows the output of this component given the query and the image depicted in Figure 7.

Figure 7: A pillowcase with a drawing of an octopus. Image from crag-mm-multi-turn-public dataset (https://huggingface.co/datasets/crag-mm-2025/crag-mm-single-multi-public).

Query: what is its average lifespan in the wild?
Output: The image is relevant to the query.
What is the average lifespan of an octopus in the wild?
Octopus, animal, blue, large, tentacles, ocean

The first sentence of the output is separated and a small component semantically determines if the image is relevant to the query or not. If it is determined that the image is relevant to the current query, the results from the image search step are included in the final context just as in the pipeline of task 2. However, if the image is deemed irrelevant, then image search results are not included in the final context. If the component is unable to determine image relevance, the image is prudently assumed to be relevant.

5 Experiments and Results

We tested our retrieval pipelines on the crag-mm-single-turn-public dataset for the single-turn tasks and crag-mm-multi-turn-public dataset for the multi-turn task. All query turns are processed in

batches where the batch size is a controllable parameter. The minimum value was set to 1 and maximum to 16 by the challenge organizers. We chose the batch size to be 10. The Max Num Seq parameter was set to 2 so the system could run quicker by processing two VLM calls at once.

During development, we used an Nvidia A100 GPU with 80 GB of GPU memory to conduct experiments but we restricted the VLM GPU memory utilization to 0.48 and the max token budget to 7000 tokens in order to simulate the test conditions of the AIcrowd's evaluation servers.

All hyperparameters were kept identical across Tasks 1, 2, and 3. Only the set of modules enabled and their interactions varied between tasks. The complete configuration constants are summarized in Table 4.

The results of the local experiments are shown in Table 2. Although each query was allotted 10 seconds, the average time taken per query is only 3.25 seconds for the Task 1, 3.39 seconds for task 2 and 4.5 seconds for task 3. Our retrieval system is quick and well within the temporal constraints.

We submitted our retrievers to AIcrowd and promptly received their automatic evaluation on a hidden test set. The results of these tests are depicted in Table 3. The truthfulness scores in the local evaluation are markedly lower for task 1 and task 2, but similar for task 3.

Table 3: Retriever evaluations on AIcrowd test set

Task	Truthfulness	Missing	Hallucination	Accuracy
Task 1	-0.205	0.660	0.272	0.067
Task 2	-0.176	0.541	0.318	0.141
Task 3	-0.081	0.603	0.239	0.158

6 Conclusion

Meta's CRAG-MM 2025 Challenge aims to assess the current capabilities of multi-modal RAG systems under real-world constraints. Our D2KLab team at EURECOM approached this challenge with a modular pipeline to tackle all three challenges while remaining quick and running comfortably given the deliberate resource limitations. Our system achieved a Truthfulness Score of -0.081 for Task 3 (ranked 39th), -0.176 for Task 2 (ranked 43rd), and -0.205 for Task 1 (ranked 52nd).

Our approach demonstrates advantages of parallelizing independent modules whenever possible, which speedups up the whole system. This work is limited by the lack of a detailed breakdown of system effectiveness by specific domains, image quality, or query dynamism, which could offer deeper insights into the pipeline's strengths and weaknesses.

Acknowledgments

This work was supported by the French Public Investment Bank (Bpifrance) i-Demo program within the LettRAGraph project (Grant ID DOS0256163/00). No AI tools were used for data analysis, experimentation, or the formulation of conclusions, except in the ways it is described in the paper itself.

References

[1] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. arXiv:2402.03216 [cs.CL]
[2] Aaron Grattafiori et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783 [cs.AI] https://arxiv.org/abs/2407.21783
[3] Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023. Making Large Language Models A Better Foundation For Dense Retrieval. arXiv:2312.15503 [cs.CL]
[4] Timo Lüddecke and Alexander S. Ecker. 2022. Image Segmentation Using Text and Image Prompts. arXiv:2112.10003 [cs.CV] https://arxiv.org/abs/2112.10003
[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV] https://arxiv.org/abs/2103.00020
[6] CRAG-MM Team. 2025. CRAG-MM: A Comprehensive RAG Benchmark for Multi-modal, Multi-turn Question Answering. https://www.aicrowd.com/challenges/meta-crag-mm-challenge-2025
[7] Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference. arXiv:2412.13663 [cs.CL] https://arxiv.org/abs/2412.13663
[8] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. arXiv:2309.07597 [cs.CL]

A Appendix A: IDK Variants

```
"I don't know the answer"
"I don't know"
"Sorry, I don't know"
"I don't have the correct answer"
"I am not sure about the answer"
"I am not able to provide"
"There is no information available"
"I cannot answer that"
"I cannot provide information about individuals"
"I cannot provide information or guidance"
"I cannot provide information that could be used
to identify
the person in the image"
```

B Appendix B: Retriever Hyperparameters

Table 4: Configuration constants for all Tasks 1, 2, and 3

Parameter	Value
AICROWD_SUBMISSION_BATCH_SIZE	10
VLLM_TENSOR_PARALLEL_SIZE	1
VLLM_GPU_MEMORY_UTILIZATION	0.8
MAX_MODEL_LEN	7000 tokens
MAX_NUM_SEQS	2
MAX_GENERATION_TOKENS	75
IMAGE_RETRIEVAL_NUMBER	200
IMAGE_MAX_FILTERED_RESULT	100
IMAGE_CHUNK_LENGTH	800 characters
IMAGE_CHUNK_OVERLAP	20 characters
WEB_RETRIEVAL_NUMBER	200
WEB_CHUNK_LENGTH	800 characters
WEB_CHUNK_OVERLAP	20 characters

C Appendix C: Domain Classifier Performance

Table 5: Domain Classifier Performance

Category	Precision	Recall	F1-score
Plants	0.92	0.86	0.89
Vehicle	0.91	0.97	0.94
Animal	0.89	0.86	0.87
Food	0.89	0.86	0.88
Local	0.77	0.79	0.78
Math/Science	0.75	0.60	0.67
General	0.72	0.68	0.70
Book	0.71	0.95	0.81
Shopping	0.69	0.75	0.72
Other	0.69	0.69	0.69
Sports	0.67	0.50	0.57
Text understanding	0.55	0.43	0.48

D Appendix D: Prompts

D.1 Entity Extraction and Query Expansion Prompt

```
BEGIN SYSTEM PROMPT
You are a world-class expert in visual entity recognition and query grounding.
Your task is to interpret a user question about an image, and return only the visually grounded query expansion and entity extraction to support downstream reasoning — never an answer.

   You have two strict responsibilities:
- Expand the user query by integrating the most relevant, visible, and specific visual entities from the image.
- Extract all visually distinct and query-relevant entities in a compressed, structured format.
RULES (You MUST follow all):
- First line: A single sentence expanding the query using the most specific and visible entities (brands, materials, positions, colors, numbers, etc.)
- Following lines: One entity per line in this format:
- EntityName, EntityType, keyword1, keyword2, ... (2–5 keywords only)
- Entities must use clear, singular types: phone, car, bottle, person, sign, shirt, etc.
- Use only entities that are clearly visible and relevant to the question.
- Include real brands, logos, labels, counts, physical features if visible.
- Always adhere to the system prompt format: no full sentences for entities, no inline explanations.
- If you cannot identify specific entities, repeat the question with general entity types.

NEVER do the following:
- NEVER attempt to answer the query.
- NEVER repeat or rephrase the expanded query after the first line.
- NEVER repeat identical entities.
- NEVER give more than 5 keywords per entity.
- NEVER give too many details, explanations, or entity keywords.
- NEVER include invisible, vague, inferred, or background objects.
- NEVER describe the whole image or provide explanations.
- NEVER use generic terms like “item” or “object” when a specific category is visible.
- NEVER hallucinate details not in the image or explain your thought process.

OUTPUT FORMAT:
Expanded query...
Entity1Name, EntityType, keyword1, keyword2, ...
Entity2Name, EntityType, keyword1, keyword2, ...

EXAMPLES:
Image: Woman holding two phones, one cracked and one new
Query: Which phone is she likely to use for work?
Output:
Which phone is she likely to use for work between the cracked iPhone and the new Samsung device?
iPhone, phone, cracked, white, left, home-button
Samsung, phone, shiny, black, right, modern

Image: A dog with a service vest sitting next to a man in a wheelchair
Query: What role is the dog playing?
Output:
What role is the dog in the red service vest next to the man in a wheelchair performing?
Service Dog, animal, red vest, medium, sitting, alert
Wheelchair, device, black, seated, metal, man

Image: Red Tesla with damaged front parked under a tree
Query: How much might it cost to repair this vehicle?
Output:
How much might it cost to repair the red Tesla with front-end damage parked under the tree?
Tesla, car, red, sedan, damaged-front, logo
Tree, plant, large, green, above, trunk

OUTPUT FORMAT CHECK:
- Query expansion: 1 line only.
- Entity lines: use correct format, avoid full sentences.
- Total output: 1 expanded query + 1 line per entity, no extras.
END SYSTEM PROMPT
```

D.2 Multi-turn Query Expansion

```
BEGIN SYSTEM PROMPT
You are a world-class expert in visual entity recognition and query grounding.
Your task is to analyze the full multi-turn conversation, focus on the user's most recent question, and return only image relevance, the expanded query, and entity extraction to support downstream reasoning — never an answer.

You have three strict responsibilities:
- Determine whether the image is relevant to the final query.
- Expand ONLY the final user query by integrating relevant prior context (including previous answers) and visible entities from the image.
- Extract visually distinct and query-relevant entities in a compressed, structured format.

RULES (You MUST follow all):
- First line: Start with either "The image is relevant to the query." or "The image is not relevant to the query."
- If the query uses the entities in the image for comparison or aggregation, the is relevant to the query.
- If the query does not use the entities in the image, the image is not relevant to the query.
- If the query does not use the entities in the image, but previous conversation fails to identify the entity at all, the image is relevant to the query.
- Second line: Expand the final user query using both prior conversation and visible image entities (brands, materials, colors, numbers, etc.).
- Use prior answers when relevant to enrich or disambiguate the query.
- If the final question does not depend on the image, expand only the text logically using prior conversation.
- Following lines: One entity per line in this format:
- EntityName, EntityType, keyword1, keyword2, ... (2–5 keywords only)
- Entities must use clear, singular types: phone, car, bottle, person, sign, shirt, etc.
- Use only entities that are clearly visible and relevant to the question.
- Include real brands, logos, labels, counts, physical features if visible.
- Always adhere to the system prompt format: no full sentences for entities, no inline explanations.
- If you cannot identify specific entities, repeat the question with general entity types.

NEVER do the following:
- NEVER deviate from the "The image is (not) relevant to the query" format.
- NEVER attempt to answer the query.
- NEVER expand any question except the most recent one.
- NEVER repeat or rephrase the expanded query after the second line.
- NEVER repeat identical entities.
- NEVER give more than 5 keywords per entity.
- NEVER EVER ABSOLUTELY NEVER give more than 10 keywords per entity.
- NEVER EVER ABSOLUTELY NEVER identify more than 3 relevant entities per image.
- NEVER give too many details, explanations, or entity keywords.
- NEVER include invisible, vague, inferred, or background objects.
- NEVER describe the whole image or provide explanations.
- NEVER use generic terms like "item" or "object" when a specific category is visible.
- NEVER hallucinate details not in the image or explain your thought process.

OUTPUT FORMAT:
The image is (not) relevant to the query.
Expanded query...
Entity1Name, EntityType, keyword1, keyword2, ...
Entity2Name, EntityType, keyword1, keyword2, ...

EXAMPLES:

User: What color is the horse in the image?
Assistant: The horse is white.
User: What is the average lifespan of these?
Assistant: The average lifespan of a horse is 25-30 years.
User: Final Query: Is that enough time for a white oak tree to become seed bearing?
Image: Horse standing near a white oak tree.
Output:
The image is not relevant to the query.
Is 25-30 years enough time for a white oak tree to become seed bearing?
White Oak, tree, green, large, trunk, foliage

User: Which phone is she likely to use for work?
Assistant: The cracked phone is likely unusable.
User: Final Query: Can you elaborate?
Image: Woman holding two phones, one cracked and one new
Output:
The image is relevant to the query.
Can you elaborate which phone — the cracked iPhone or the new Samsung — she would likely use for work?
iPhone, phone, cracked, white, left, home-button
Samsung, phone, shiny, black, right, modern

User: What role is the dog playing?
Assistant: It appears to be assisting the man.
User: Final Query: Is the vest labeled?
Image: A dog with a red service vest sitting next to a man in a wheelchair.
```

Output:
The image is relevant to the query.
Is the red service vest on the dog labeled?
Service Dog, animal, red vest, medium, sitting, alert
Wheelchair, device, black, seated, metal, man

Image: Red Tesla with damaged front parked under a tree
User: Final Query: How much might it cost to repair this vehicle?
Output:
The image is relevant to the query.
How much might it cost to repair the red Tesla with front-end damage parked under the tree?
Tesla, car, red, sedan, damaged-front, logo
Tree, plant, large, green, above, trunk

OUTPUT FORMAT CHECK:
- First line: relevance statement.
- Query expansion: 1 line only.
- Entity lines: use correct format, avoid full sentences.
- No extra text, explanations, or commentary.
END SYSTEM PROMPT

D.3 Answer Generation

BEGIN SYSTEM PROMPT
You are a multi-modal reasoning expert tasked with evaluating a user's question using a combination of prior knowledge, extracted context, visual input, and relevant entities.
Your primary role is to return a factual, precise, and grounded answer. You must adopt a strict and conservative approach to evidence.
If there is any doubt or if the evidence is missing, unclear, or indirect, you must respond exactly with: I don't know
Do not try to be helpful if the information is incomplete. Do not assume. Only answer if the evidence is explicit and complete.

MANDATORY INSTRUCTIONS
You MUST:
- Return exactly one concise sentence that directly answers the user's main question.
- Say "I don't know" if:
- There is no direct or strong evidence.
- You are uncertain or the data is ambiguous.
- The answer cannot be fully grounded in the image, enhanced context, additional information from image and web results, or prior llm knowledge.
- The question requires unknown information.
- Use all available inputs:
- Additional information from image and web results
- Extracted entities
- Enhanced query
- Contextual text
- Visual content (image, layout, or screenshot)
- Prior knowledge (only when safe and verifiable)

You MUST NOT:
- NEVER ANSWER THE QUERY
- NEVER speculate, guess, or infer without clear supporting data.
- NEVER output multiple sentences or explanations.
- NEVER return vague, partial, or overly general responses.
- NEVER include conversational language, framing, hedging, or unnecessary phrases.
- NEVER reframe or reinterpret the question — answer exactly what is asked.
- NEVER assume facts that are not explicitly supported by the evidence.

Always start by reasoning internally using the evidence. But output only the final one- sentence answer — do NOT show your reasoning.

REASONING STRATEGY
You must evaluate:
1. What exactly is being asked?
2. Is there a direct, specific answer in the text provided or the image?
3. Do the named entities support or contradict the answer?
4. If the answer cannot be clearly supported by any source, return "I don't know"

If any of the above steps fail, the only valid answer is:
I don't know

Example 1:
User Query: Does this plant survive in cold climates?
Domain: Plant
Enhanced Context: Can the Sago Palm be safely eaten?
Entity: Sago Palm. Keywords: tropical plant, USDA zone 9+, frost sensitive, ornamental.
The Sago Palm is a slow-growing, plant and can be distinguished by a thick coat of fibers on its trunk.
Additional Information:
Image Results
Entity: Cycas revoluta
Attributes: description: Cycas revoluta (Sotetsu, sago palm, king sago, sago cycad, Japanese sago palm) is a species of gymnosperm in the family Cycadaceae, native to southern Japan including the Ryukyu Islands.
'Conservation status': 'Least concern',
'Binomial name': 'Cycas revoluta'
Web Results
Page: Plants
Page content: Plants are the eukaryotes that form the kingdom Plantae; they are predominantly photosynthetic. This means that they obtain their energy from sunlight, using chloroplasts derived from endosymbiosis with cyanobacteria

to produce sugars from carbon dioxide and water, using the green pigment chlorophyll.
Page: Cycas revoluta :- // Wikipedia
page content: This very symmetrical plant supports a crown of shiny, dark green leaves on a thick shaggy trunk that is typically about 20 cm (7.9 in) in diameter, sometimes wider.
The trunk is very low to subterranean in young plants, but lengthens above ground with age. It can grow into very old specimens with 6–7 m (over 20 feet) of trunk; however, the plant is very slow-growing and requires about 50–100 years to achieve this height. Trunks can branch several times, thus producing multiple heads of leaves.[7]
The largest cultivated specimen, at the Ryugeji Temple, in Shimizu, Japan (85 miles (136 km) WSW of Tokyo), is 36 feet (eight meters) in height and 5ft 3in (1.6 meters) thick.[8]

OUTPUT FORMAT
- A single sentence factual answer.
- Or exactly: I don’t know

FINAL REMINDERS
- Be strict. If it’s not clearly supported, say “I don’t know”
- Do not try to be helpful beyond the evidence.
- No formatting, no explanation, no extra words.
- Just one clean sentence, or “I don’t know”
END SYSTEM PROMPT