

# ***PuppyCompanion*** 🐶

## **Task 1: Defining your problem and audience**

### **1. Concise problem description**

New puppy owners are faced with a mass of disparate and sometimes conflicting information regarding their pet's care, training, and health, creating anxiety and uncertainty during an already stressful period of mutual adjustment.

### **2. Why this is a problem for your specific user**

Puppy owners, especially first-time puppy owners, quickly find themselves overwhelmed by the multitude of responsibilities and decisions to ensure the well-being and upbringing of their new pet.

Faced with the abundance of information available online, in books, and from those around them, they struggle to discern reliable and relevant advice for their specific situation. This information overload generates stress and anxiety, precisely at a time when they are trying to establish a harmonious relationship with their puppy.

The lack of a centralized, reliable, and personalized source of information forces these owners to navigate multiple, fragmented resources, spending considerable time searching for answers to their urgent questions (behavior, health, nutrition, training).

This constant search for information reduces the quality time spent with the animal and can lead to errors of judgment with potentially harmful consequences for the puppy's behavioral and physical development. Furthermore, the lack of personalized generic advice doesn't take into account the specifics of the puppy's breed, age, or environment, sometimes leading to the application of methods unsuitable for their particular situation.

I know exactly what I'm talking about, having been the proud owner of a purebred Australian Shepherd puppy for 15 days.

Despite all my prior reading and the numerous videos I've watched, I find myself daily confronted with billions of questions without clear answers and struggle to find sound advice for everyday activities.

I quickly realized that properly raising a puppy is truly a full-time job during the first few months, a crucial period for its future development and decisive for its entire life. This personal experience confirms the crucial need for a tailored solution to support owners like me through this critical phase.

## Task 2: Propose a solution

### 1. Description of the proposed solution

I propose "PuppyCompanion," an intelligent assistant for new puppy owners that leverages a multi-agent architecture to provide reliable and personalized advice. The application will use a sophisticated routing system to direct questions to specialized knowledge sources, ensuring accurate answers in various areas such as puppy training, health concerns, behavioral issues, and product recommendations.

Users will interact with PuppyCompanion via a conversational interface where they can ask any dog-related question. Under the hood, the system will first determine if the query is dog-related and then route it to a RAG system that retrieves information from reliable contextual resources (specialized books, puppy video transcripts, training videos, etc.).

For this challenge, I am using only one book as a reference: "Puppies for Dummies," a popular reference work on the subject.

As the project will progress, particularly for the demo day, I plan to expand the reference content type and, if time permits, include transcripts of popular YouTube videos from major players in this field.

If the RAG system's response is deemed insufficient, that is the system is unable to find specific context in its reference database, it will automatically switch to web search results via the Tavily API.

This dual-source approach ensures comprehensive coverage while maintaining factual accuracy and security, particularly for health-related questions. The system will also intelligently recognize when questions fall outside its area of expertise (non-canine topics) and politely redirect users, ensuring they always receive appropriate advice.

#### Tools used in each part of the stack

- **LLM:**  
OpenAI's GPT-4o-mini for its balance between performance and cost-effectiveness. This model powers the application's various components, including the intelligent router, response evaluation, and final response generation, ensuring consistency of tone and reasoning capabilities throughout the system.
- **Embedding Model:**  
A combination of embedding approaches, with OpenAI's text-embedding-3-small as the primary model. For specialized domains, the Snowflake Arctic Embed model offers an alternative embedding strategy with domain-specific capabilities.

- **Orchestration:**  
LangGraph orchestrates the multi-agent workflow, providing a structured graph-based approach to managing state and routing between different components. This allows for a clear separation of routing logic, knowledge retrieval, evaluation, and response generation.  
Qdrant Vector Database for storing and querying document embeddings, chosen for its high performance in similarity search and its ability to handle both in-memory and persistent storage, making it ideal for development and production environments.
- **Monitoring:**  
Comprehensive logging strategy across the application to track execution flow, router decisions, and response quality. This will be extended in production to include error tracking and analytics.
- **Evaluation:**  
RAGAS framework for systematic application evaluation, with custom metrics including Fidelity, Contextual Recall, Answer Relevance, Entity Recall, and a "home made" Domain Detection Accuracy metric to ensure the system correctly identifies dog-related versus non-dog-related questions.
- **Serving & Inference:**  
Initial deployment on Hugging Face to enable easy and public access to the application. This approach provides a ready-to-use solution with an integrated web interface, while also making it easy to share the model and its capabilities with the community. In later phases, we may consider migrating to containerized services for a more robust production deployment.

## 2. Use of Agents and Agentic Reasoning

The solution employs a sophisticated multi-agent architecture orchestrated via LangGraph. At its heart is a "smart router" agent that analyzes incoming questions to determine if they are dog-related, using reverse logic for more reliable classification (by asking if the question is NOT dog-related). This reverse approach yields better discrimination results for the LLM in charge of this analysis. This routing is crucial because it directs the workflow either to specialized knowledge sources or politely informs users when questions fall outside the system's scope.

For dog-related questions, the system first employs a RAG-based agent that retrieves and synthesizes information from the knowledge base. What makes this approach particularly powerful is the evaluation agent, which reviews RAG answers according to quality criteria (checking for vague advice, nonspecific answers, or redirects). When answers are deemed

unsatisfactory, the system automatically switches to a Tavily search agent, which provides web-based information as a secondary knowledge source.

### 3. Agentic reasoning is most evident in critical components

1. **Domain detection logic**, which uses reverse psychology and clear examples to make binary classification decisions more robust
2. **Answer quality assessment**, which analyzes answers according to multiple criteria to determine their sufficiency
3. **Final answer synthesis**, which intelligently combines information from multiple sources (RAG and web search) into coherent and useful advice

This multi-agent architecture allows for graceful degradation, that is, if the primary knowledge source fails, the system can still provide useful information from alternative sources, rather than simply saying "I don't know."

The design also allows for future expansion by adding specialized agents for specific areas of puppy care (training, health, nutrition) without disrupting the overall architecture.

## Task 3: Dealing with the data

### 1. Data sources and external APIs

To create an effective and reliable PuppyCompanion application, I carefully select a diverse set of data sources and APIs. Each choice will be made considering the specific needs of puppy owners and the necessity to provide accurate, up-to-date, and relevant information.

#### Primary Document Base for RAG

##### 1. Puppies For Dummies (PDF)

- **Why:** This comprehensive manual (which will be the first source of context integrated into our code), offers a solid foundation of structured knowledge. I chose this resource because it systematically covers all essential aspects of puppy education, from behavior to daily care.
- **Usage:** Primary source for our RAG system, providing reliable and consistent foundations.

##### 2. American Veterinary Documentation

- **American Kennel Club (AKC):** Articles on breeds, development, and education
- **American Veterinary Medical Association (AVMA):** Resources on health and preventive care
- **VCA Animal Hospitals:** Clinical articles reviewed by veterinarians
- **Why:** These organizations are undisputed references in the canine field, offering scientifically validated information. Institutional sources guarantee greater reliability than blogs or forums.
- **Usage:** Enrichment of our knowledge base on health issues and breed-specific needs.

##### 3. Resources from Recognized Dog Trainers

- Works by Jean Donaldson, Sophia Yin, Patricia McConnell, and Ian Dunbar
- **Why:** These behaviorists are recognized for their science-based and positive reinforcement approaches. Their inclusion balances our content between health and behavior.
- **Usage:** Information source for complex education and behavior questions.

##### 4. Educational YouTube Channel Content

- Zak George's Dog Training Revolution, Kikopup, McCann Dog Training
- **Why:** These content creators visually demonstrate training techniques, offering practical and accessible advice that books cannot always effectively communicate.
- **Usage:** Transcriptions to enrich our knowledge base on practical and visual training techniques.

## External APIs

### 1. Tavily Search API

- **Why:** This API provides an easy way to search on the web. I will configure a limit of 5 results per search to balance depth and relevance.
- **Usage:** Fallback mechanism that will be activated when the response quality of our RAG knowledge base is insufficient.

### 2. The Dog API (<https://thedogapi.com/>)

- **Why:** This API provides structured data on over 350 breeds, allowing precise customization of advice according to the specific breed of the puppy.
- **Usage:** Enrichment of responses with breed-specific information, such as exercise needs, medical predispositions, and personality traits.

## 2. Default Chunking Strategy

For this challenge, I simply feed my knowledge base with a PDF book containing unstructured content. It contains text, boxes, diagrams, photos, and more. There is plenty to clean and reformat to be able to use it in our retrieval and context enrichment tool. Therefore I choose the Unstructured.io library for this task.

It is incredibly effective at organizing in a hierarchical way contents as title, illustration, narrative text, and more.

By using the generated metadata, we can then filter the elements that contain the most useful content for our problem. experimenting with different configurations.

Once the content filtered, I chose these parameters:

- **Maximum chunk size:** 1800 characters
- **New creation after:** 1500 characters
- **Overlap:** 400 characters

### Why these specific parameters?

1. **Maximum size (1800 characters):** Tests showed that this length (~300-350 words) represents an optimal balance for my domain. Shorter chunks (500-1000 characters) excessively fragmented step-by-step training procedures, while longer chunks (>2500 characters) diluted semantic relevance and retrieved too much non-essential information.
2. **New creation after 1500 characters:** This value slightly lower than the maximum size ensures that we don't create chunks systematically reaching the upper limit, which could artificially truncate natural sections of the text.
3. **Overlap of 400 characters (20-25%):** This level of overlap is crucial to maintain consistency in chunk content. Low overlap (100-200 characters) results in information loss at our chunk boundaries, while high overlap (>500 characters) creates excessive redundancy without improving relevance.

I also do :

- a filtering of short fragments (<50 characters) to eliminate non-informative elements

- some text cleaning (removing figure references, correcting special characters)
- a **preservation of structural metadata** (distinction between narrative text, lists, and titles)

### 3. Specific Data for Other Parts of the Application

#### Embedding model Fine-tuning plan

For my fine-tuning strategy, I will build a specialized dataset from our document base:

- **Dataset creation:** I will randomly select 110 chunks from our processed document base to ensure a representative sample of topics
- **Question generation:** For each selected chunk, I will construct corresponding questions that a puppy owner might naturally ask, ensuring semantic diversity
- **Training structure:** I will organize these into contrastive triplets (question, relevant passage, irrelevant passage)

**Why this approach is necessary:** Generic embedding models struggle to capture canine terminological nuances. For example, terms like "nipping" vs "biting" or "separation anxiety" vs "hyper attachment" have distinct clinical implications that our fine-tuning will help differentiate effectively.

The modest size of 110 samples strikes a balance between having sufficient examples to improve domain-specific understanding while remaining practical for implementation without extensive computational resources.

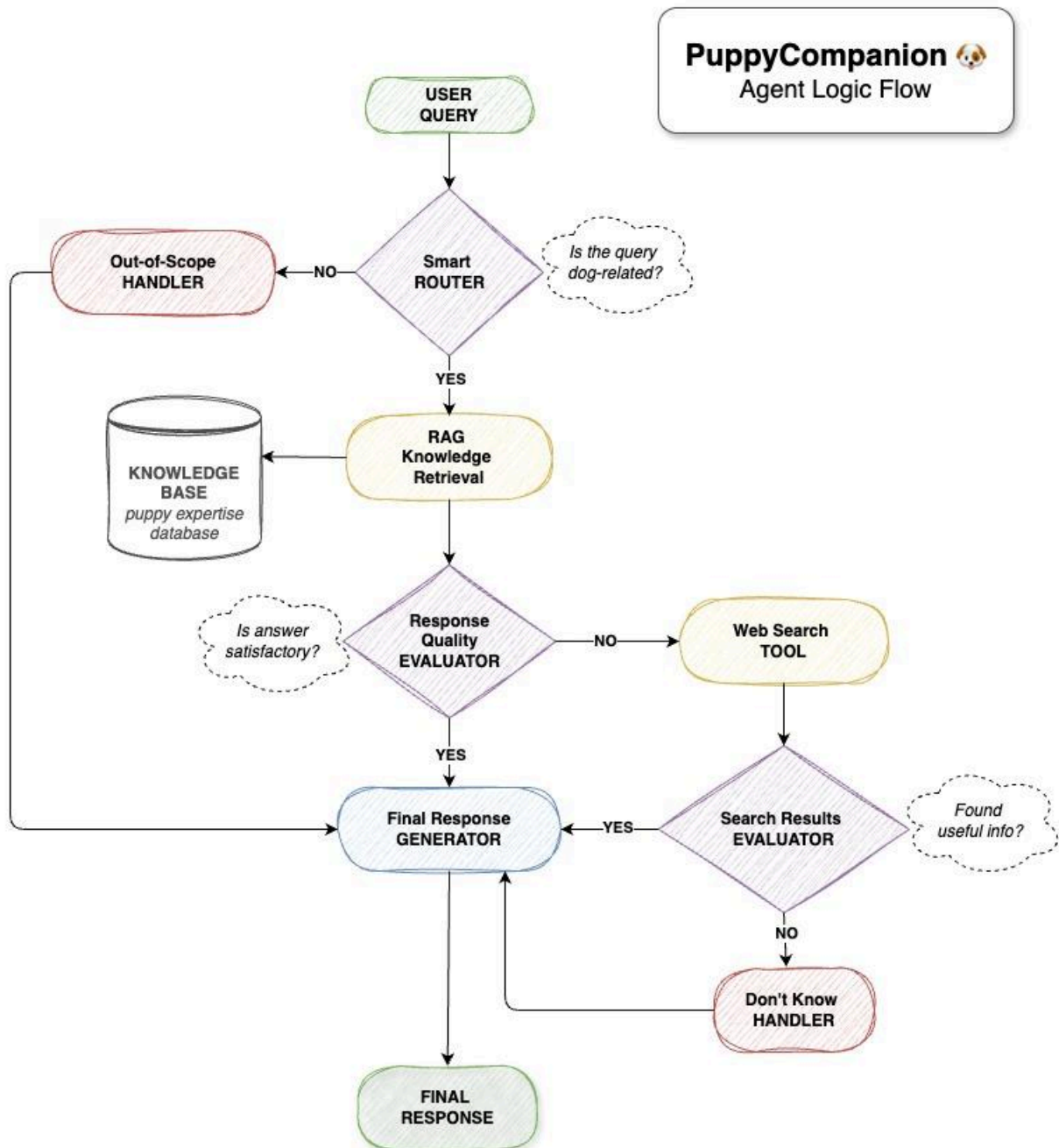
The Github repository of the project is here :

[https://github.com/JTh34/AIE6-Julien/tree/main/11\\_Midterm\\_Challenge](https://github.com/JTh34/AIE6-Julien/tree/main/11_Midterm_Challenge)

## Task 4: Building a quick end-to-end prototype

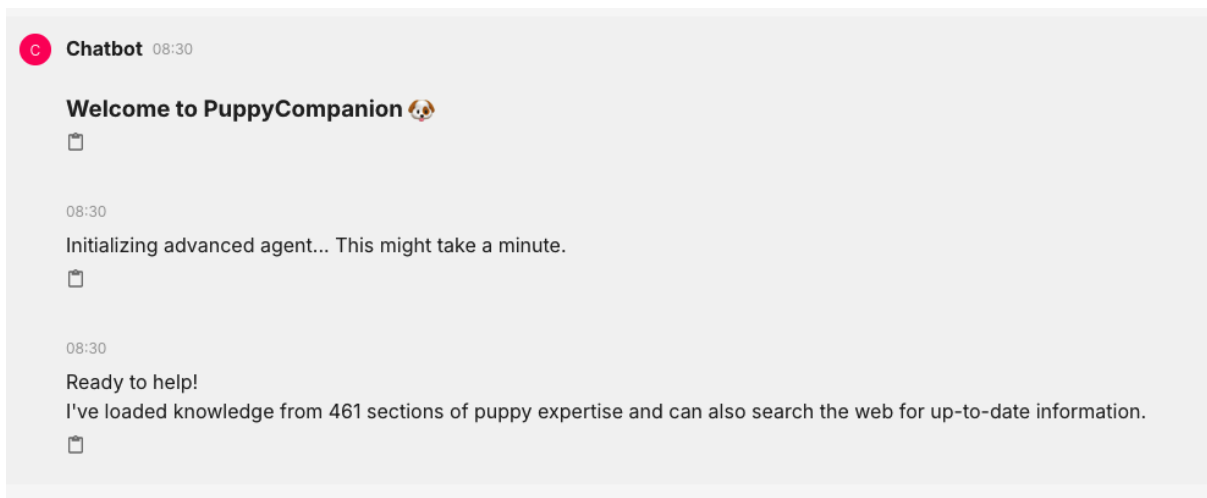
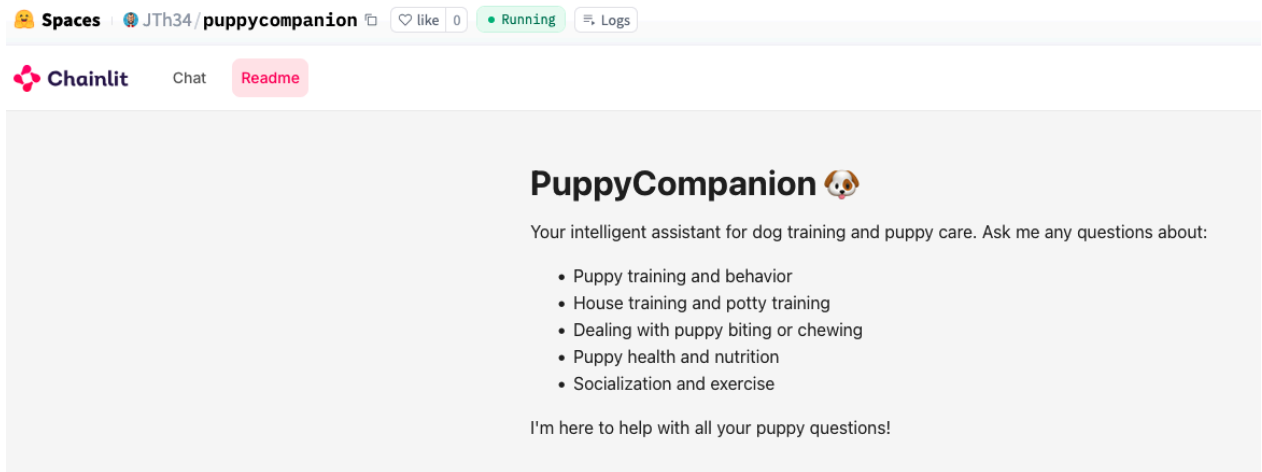
My end-to-end RAG application is running here :

<https://huggingface.co/spaces/JTh34/puppycompanion>





Some screenshots of the main application states:



**You** 08:32

How do I teach my puppy to do his business outside?

**Chatbot** 08:32

Searching for the best answer...



08:33

To teach your puppy to do their business outside, follow these steps:

1. Choose a Potty Area: Designate a specific outdoor spot (like grass or mulch) where you want your puppy to go, ideally about 10 to 20 feet from your door.
2. Establish a Routine: Take your puppy to the potty area regularly, especially after eating, drinking, waking up, or playing. Use a consistent cue phrase like "Go outside."
3. Wait for Them to Eliminate: Once at the spot, wait for your puppy to go potty. You can use a phrase like "Get busy" to encourage them.
4. Positive Reinforcement: After your puppy eliminates, celebrate with praise, play, or treats to reinforce the behavior.
5. Teach a Signal: Encourage your puppy to signal when they need to go out, such as ringing a bell.
6. Gradually Decrease Assistance: As your puppy learns the routine, allow them to go to the potty area on their own.
7. Be Patient and Consistent: House-training takes time. Stay positive, and if accidents happen, clean them up calmly while continuing to reinforce the correct behaviors.

By being consistent and patient, your puppy will learn to do their business outside effectively.

[RAG Tool - Response from the book "Puppies for Dummies"]



**You** 08:34

How do I train my puppy to detect landmines?

**Chatbot** 08:34

Searching for the best answer...



08:34

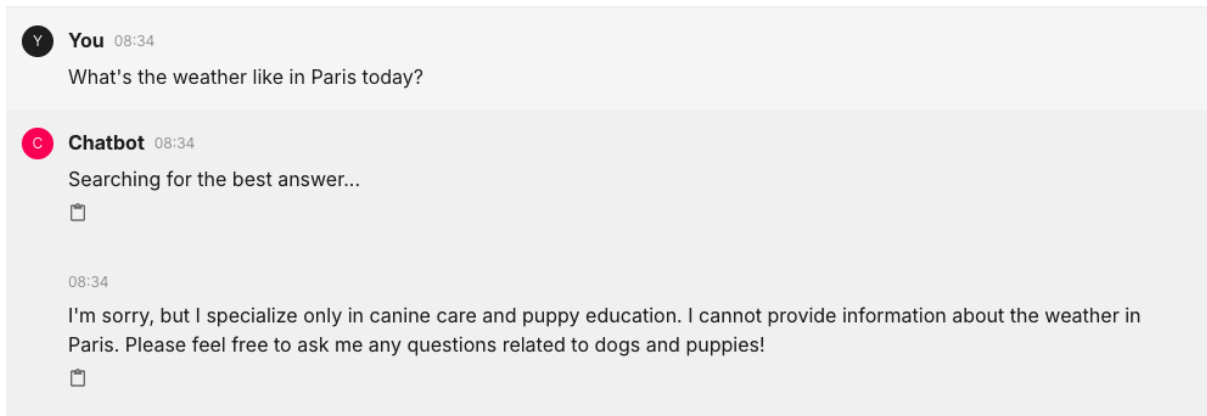
Training a puppy to detect landmines involves specialized programs that focus on teaching the dog to locate the scent of explosives. The training typically includes the following key components:

1. Scent Identification: Dogs are trained to recognize the specific odors of explosives, such as 2,4,6-TNT. This foundational step is crucial for their ability to detect landmines.
2. Alerting Behavior: Once the dog detects the scent, they are trained to perform a specific action, such as sitting still, to alert their human partner to the location of the mine.
3. Safety Protocols: Training also emphasizes safety for the dog, ensuring they know how to stay safe while marking the spot of a discovered mine.
4. Consistent Practice: Ongoing training and practice help reinforce the dog's skills and ensure they remain effective in detecting landmines.

Programs like those offered by the Marshall Legacy Institute provide structured training for mine detection dogs, focusing on these essential skills.

[Tavily Tool - Response from the Internet]





The Github repository of the application is here :

[https://github.com/JTh34/AIE6-Julien/tree/main/11 App Challenge/puppycompanion-app](https://github.com/JTh34/AIE6-Julien/tree/main/11_App_Challenge/puppycompanion-app)

## Task 5: Creating a “Golden test dataset”

### 1. Golden Dataset Generation and Composition

I created a sophisticated golden test dataset using the CanineTestsetGenerator, which leverages RAGAS Test Set Generator and GPT-4o-mini to generate 12 domain-specific questions covering puppy training, nutrition, behavior problems, and healthcare, plus 2 deliberately included out-of-domain questions to test boundary recognition.

### 1. RAGAS Evaluation Results

Metric	Score
Context Recall	0.39
Faithfulness	0.92
Answer Relevancy	0.88
Context Entity Recall	0.34
Domain Detection Accuracy	0.93

## RAGAS Metrics

- **Context Recall** : Measures how completely the system retrieves relevant information from the knowledge base.
- **Faithfulness** : Evaluates whether responses stay true to retrieved context without hallucinations.
- **Answer Relevancy** : Assesses how directly responses address the user's specific query
- **Context Entity Recall** : Measures retrieval of specific named entities relevant to the query.

## Custom Metric

**Domain Detection Accuracy** : My custom metric to evaluate the ability of the system to distinguish dog-related from unrelated queries. It works by first classifying each question as dog-related or not, then checking if the response of the system was appropriate.

This tests our "smart router" which uses reverse psychology (asking if questions are NOT dog-related) to improve classification accuracy.

## 3. Performance Analysis and Conclusions

### Strengths:

- Our system demonstrates excellent faithfulness .
- The high answer relevancy (0.88) confirms that responses are well-aligned with user queries, providing appropriate and targeted information.
- The high domain detection accuracy (0.93) demonstrates our "smart router" efficiently distinguishes canine-related questions from unrelated queries, maintaining the system's focus and reliability.

### Areas for improvement:

The relatively low context recall (0.39) and context entity recall (0.34) suggest our retrieval mechanism is not capturing all relevant information from the knowledge base.

### Hybrid pipeline considerations:

- Evaluating this hybrid system presented unique challenges, requiring separate assessment of RAG-sourced and web-searched responses.
- Our evaluation methodology categorized responses based on source and applied appropriate metrics to each category, providing nuanced performance insights across different response mechanisms.

# Task 6: Fine-tuning open-source embeddings

## 1. Fine-tuned embedding model implementation

I fine-tuned an embedding model on pairs of Question/Answers to better understand canine-specific terminology and semantic relationships in puppy care questions. The original OpenAI embedding model has been replaced with a domain-specialized Snowflake Arctic Embed model that's been fine-tuned on our puppy care corpus.

## Fine-tuning process overview

Starting with the base model **Snowflake/snowflake-arctic-embed-l**, I fine-tuned it using a dataset generated from our "Puppies For Dummies" PDF. The process included:

1. Creating a specialized dataset of 150 examples split into 105 training, 22 validation, and 23 test examples
2. Generating natural puppy owner questions for each document chunk using GPT-4o-mini
3. Training for 5 epochs using MatryoshkaLoss with multiple embedding dimensions (1024, 512, 256, 128, 64)
4. Regular evaluation on our validation set to track improvement

## Performance improvements

The fine-tuning produced significant retrieval improvements across the 5 epochs of training:

- **Accuracy@1**: Improved from 54.55% to 72.73% (+18.18%) a
- **MRR@10**: Improved to 0.8235
- **NDCG@10**: Reached 0.8667
- **Recall@3**: Increased to 86.36%

These metrics indicate our model is now much better at retrieving the most relevant document for puppy-related queries, which should directly address our previously identified weakness in context recall (0.39).

[265/265 08:11, Epoch 5/5]																	
Step	Training Loss	Validation Loss	Cosine Accuracy@1	Cosine Accuracy@3	Cosine Accuracy@5	Cosine Accuracy@10	Cosine Precision@1	Cosine Precision@3	Cosine Precision@5	Cosine Precision@10	Cosine Recall@1	Cosine Recall@3	Cosine Recall@5	Cosine Recall@10	Cosine Ndcg@10	Cosine Mrr@10	Cosine Map@100
50	No log	No log	0.545455	0.818182	0.909091	1.000000	0.545455	0.272727	0.181818	0.100000	0.545455	0.818182	0.909091	1.000000	0.775166	0.702273	0.702273
53	No log	No log	0.545455	0.818182	0.909091	1.000000	0.545455	0.272727	0.181818	0.100000	0.545455	0.818182	0.909091	1.000000	0.775166	0.702273	0.702273
100	No log	No log	0.590909	0.772727	0.909091	1.000000	0.590909	0.257576	0.181818	0.100000	0.590909	0.772727	0.909091	1.000000	0.780848	0.711364	0.711364
106	No log	No log	0.681818	0.772727	0.909091	1.000000	0.681818	0.257576	0.181818	0.100000	0.681818	0.772727	0.909091	1.000000	0.819311	0.763312	0.763312
150	No log	No log	0.681818	0.863636	0.954545	1.000000	0.681818	0.287879	0.190909	0.100000	0.681818	0.863636	0.954545	1.000000	0.843933	0.793182	0.793182
159	No log	No log	0.681818	0.863636	0.954545	1.000000	0.681818	0.287879	0.190909	0.100000	0.681818	0.863636	0.954545	1.000000	0.843933	0.793182	0.793182
200	No log	No log	0.727273	0.863636	0.954545	1.000000	0.727273	0.287879	0.190909	0.100000	0.727273	0.863636	0.954545	1.000000	0.860709	0.815909	0.815909
212	No log	No log	0.727273	0.863636	0.954545	1.000000	0.727273	0.287879	0.190909	0.100000	0.727273	0.863636	0.954545	1.000000	0.866660	0.823485	0.823485
250	No log	No log	0.727273	0.863636	0.954545	1.000000	0.727273	0.287879	0.190909	0.100000	0.727273	0.863636	0.954545	1.000000	0.866660	0.823485	0.823485
265	No log	No log	0.727273	0.863636	0.954545	1.000000	0.727273	0.287879	0.190909	0.100000	0.727273	0.863636	0.954545	1.000000	0.866660	0.823485	0.823485

2025-05-12 14:43:00,378 - INFO - Information Retrieval Evaluation of the model on the dataset in epoch 0.9433962264150944 after 50 steps:  
 2025-05-12 14:43:04,976 - INFO - Queries: 22  
 2025-05-12 14:43:04,979 - INFO - Corpus: 12

My fine-tuned embedding model is publicly available at:

<https://huggingface.co/JTh34/puppy-embed-8985966a>

## 2. RAGAS Evaluation Results

Metric	Score
Context Recall	0.49
Faithfulness	0.92
Answer Relevancy	0.88
Context Entity Recall	0.41
Domain Detection Accuracy	0.86

## Task 7: Assessing performance

### 1. How performance compares to the original RAG application

The comparative evaluation between our original OpenAI embedding model and the fine-tuned Snowflake model reveals significant improvements in information retrieval capabilities:

Metric	OpenAI Embeddings	Snowflake Fine-tuned	Change
Context Recall	0.39	0.49	+25.6%
Faithfulness	0.92	0.92	0%
Answer Relevancy	0.88	0.88	0%
Context Entity Recall	0.34	0.41	+20.6%
Domain Detection Accuracy	0.93	0.86	-7.7%

### Result analysis

Our fine-tuned model shows improvement precisely in the areas where we needed it most:

- **Context Recall (+25.6%):** The improvement means that the system now retrieves significantly more relevant information to answer questions.
- **Context Entity Recall (+20.6%):** This improvement shows that the system better captures specific entities (breeds, training techniques, medical terms) mentioned in questions, which is important for precise advices.
- **Quality Metrics:** Faithfulness (0.92) and Answer Relevancy (0.88) scores remained constant, indicating that the retrieval improvements did not compromise response quality and accuracy.
- **Domain Detection Accuracy (-7.7%):** This reduction in ability to distinguish canine questions from other topics remains acceptable given the significant gains in information retrieval.



The validation data during fine-tuning further confirms these improvements, with an increase in Accuracy@1 from 54.55% to 72.73%, demonstrating that the model is now much more effective at matching a question to its most relevant context.

What is really interesting in the context of our project is that the fine-tuning training was done in a reasonable way (20 min on the GPU of my MacBookPro M4) with relatively few samples (150 Question/Answer pairs in total). All with an open-source model. This can be very appreciable in terms of costs.

## 2. Planned improvements for the second half of the course

By the demo day, I plan to enhance the current application with **a more diversified knowledge base** focusing on training techniques, particularly clicker training, and canine health, with special emphasis on understanding dogs' psychological and physiological needs while also providing guidance for managing minor everyday health issues.

To accomplish this, I will enrich my knowledge base with highly specialized books on these subjects.

I also plan to add **a planning agent** that will provide temporal and progressive programs to address specific problems or topics (such as step-by-step house training, teaching a puppy to stop nipping, etc.)

Additionally, I intend to integrate **a specialized agent dedicated to creating step-by-step programs**. This agent will be capable of breaking down any task or objective into a precise list of actions to implement based on available resources (location, duration, financial means, dog breed, living environment).

Of course, this entire ecosystem (the current application enhanced by these new agents) will be **supervised and orchestrated by a central supervisor agent** and potentially **a content validation agent** that ensures the quality of output responses.