autoMATE

AUTOMOTIVE MULTIMODAL AUGMENTED TECHNICAL EXPERT

Vision-Enhanced RAG: Next-Gen Interaction with Car Manuals

Arnur Nurakhmetov

Gabriele Rizzo

Emiliano Simonelli

Sara Silva

LUISS

SUMMARY

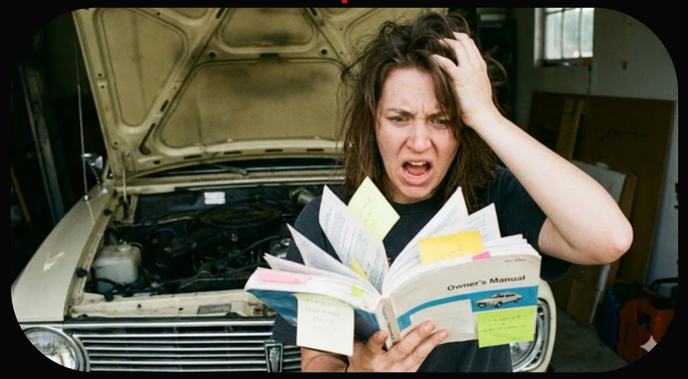
LUISS

The problem

Have you ever read your car's manual to solve a problem or to obtain informations about an unknown button?

Automotive manuals are often long, dense, and difficult to search.

Technical diagrams, tables, and maintenance procedures are hard to interpret, and traditional search fails to retrieve the exact information needed.



Our Vision:
Ask Naturally, Get Instant Answer

An AI-powered assistant that allow any user to query manuals in natural language while receiving precise instructions and relevant visual references.

LUISS

Solution Approach

PDF-Based RAG Pipeline:

- Direct ingestion of official PDF manuals using advanced OCR.
- Extracts both textual content and visual elements while preserving structure.

Multi-Modal Architecture:

- The system understands and retrieves both Text and Images.
- Combines semantic text search with visual understanding.

Integration of Images:

- Technical manuals are visually dense (diagrams, icons, layouts).
- Text-only search misses critical context.

Precision & Context:

- Vehicle-aware filtering prevents mixing information between different car models.

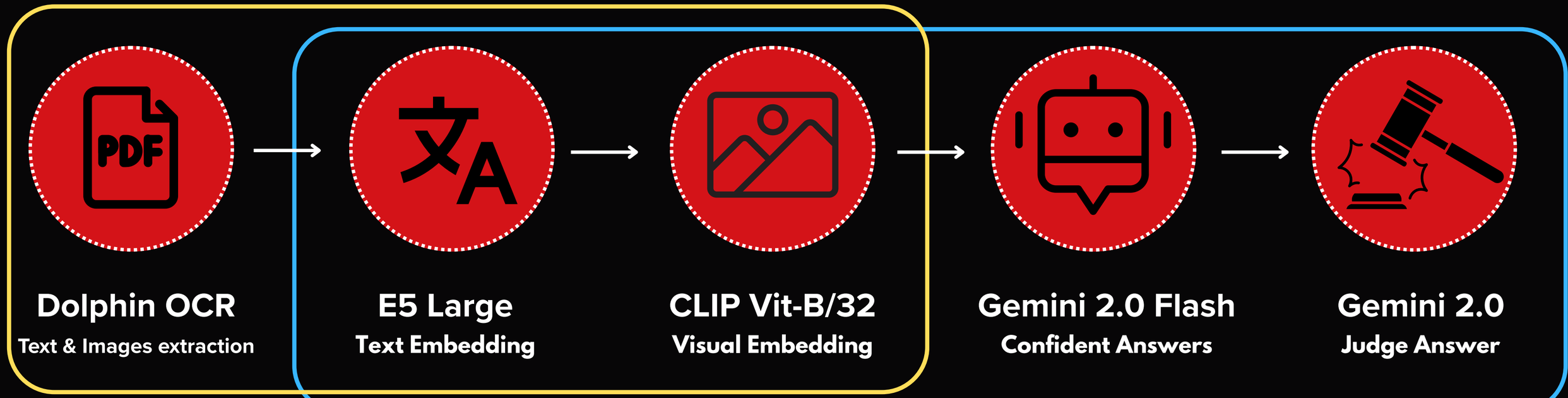
Traditional	Our System
Time to answer 15+ minutes	Time to answer 1.2 seconds
Text only	Text + Images
Keyword matching	Semantic search
No citations Miss diagrams	Page references Visual matching

→ 750x faster, multi-modal

LUISS

Technical architecture 1/4

Five Models, One Intelligent Pipeline



LUISS

Technical architecture 2/4

1. Manual Ingestion (PDF Input):

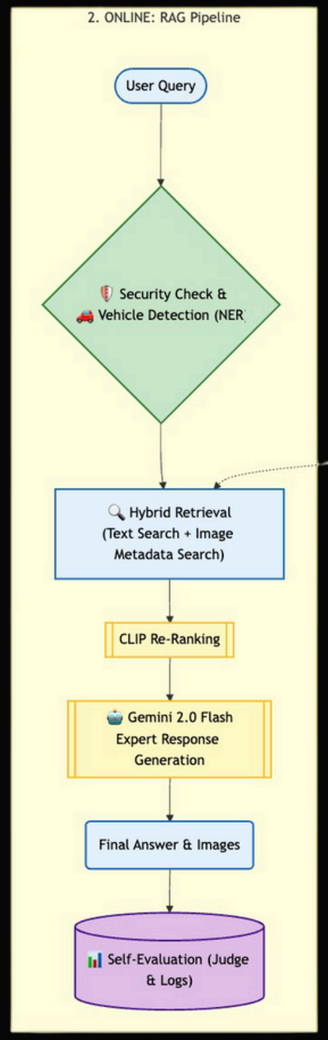
- Import car manual PDF.
- Pipeline triggers the extraction stage.

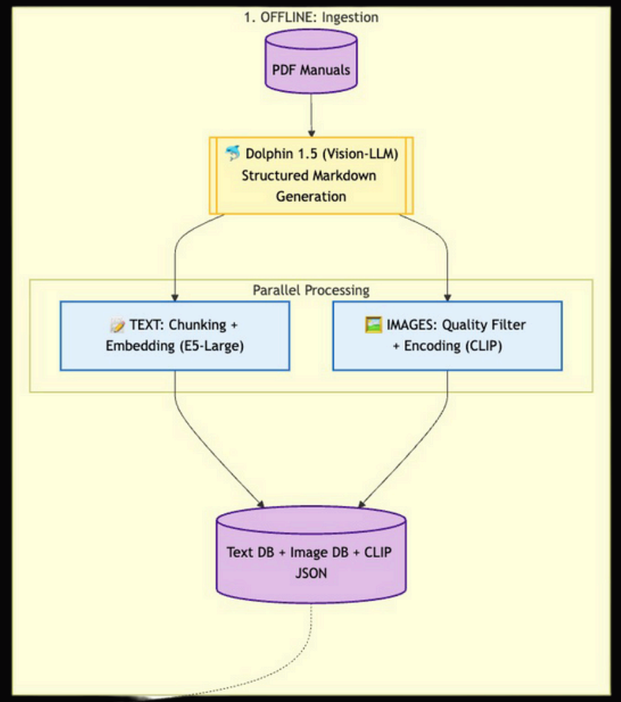
2. Dolphin OCR - Text & Image Extraction:

- Vision LLM → pdf as images.
- Extracts paragraphs, tables, captions.
- Outputs: structured Markdown file.

3. Text Embedding (E5-L):

- Text chunking.
- Embedding → Semantic Vector





4. Image Embedding (CLIP ViT-B/32):

- Visual embedding of each extracted image

5. Vector Database Creation (ChromaDB):

- Stores text vectors.
- Stores image vectors.
- Enables fast similarity search.

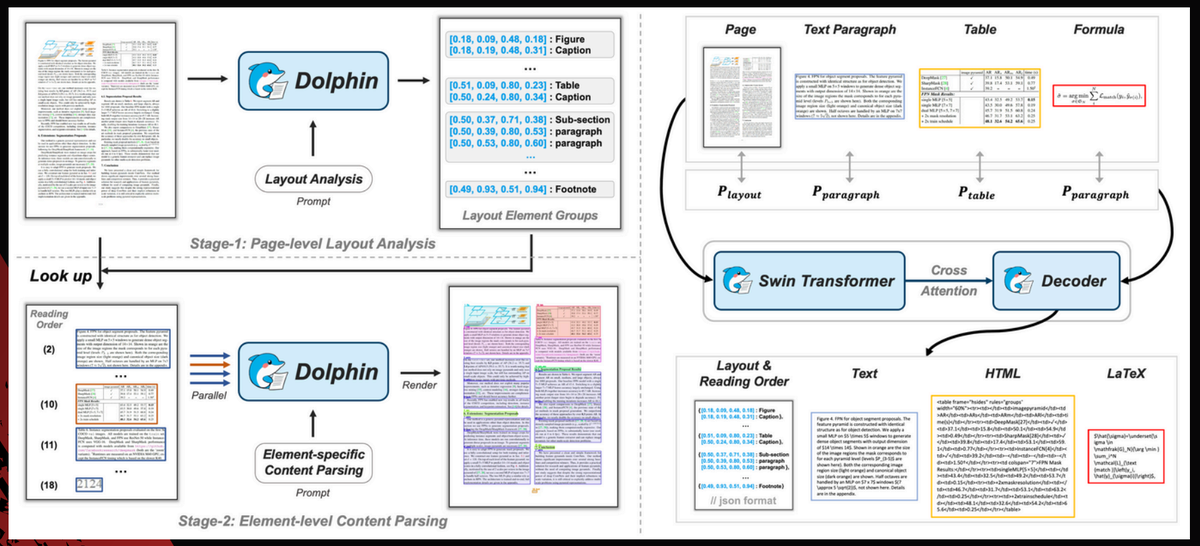
6. LLM - Gemini 2.0 Flash

- Both answer generator and judge

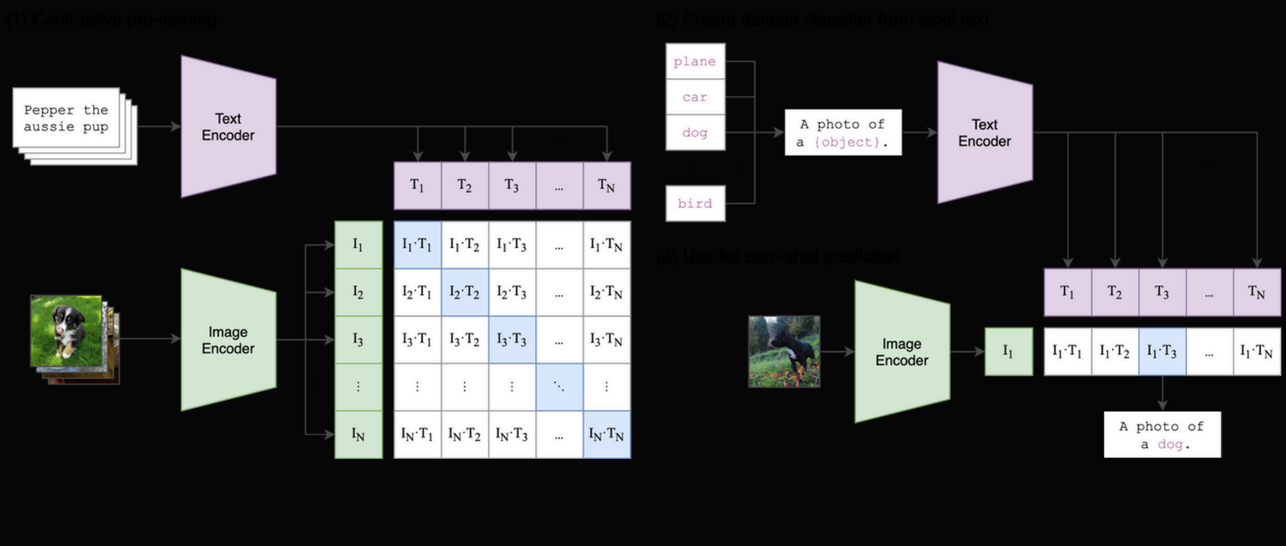
LUISS

LUISS

Dolphin + CLIP: The winning combination



https://github.com/bytedance/Dolphin



https://github.com/openai/CLIP

LUISS

Technical architecture 3/4

(1) The Core Engine: Dolphin 🐬 (Vision-LLM)

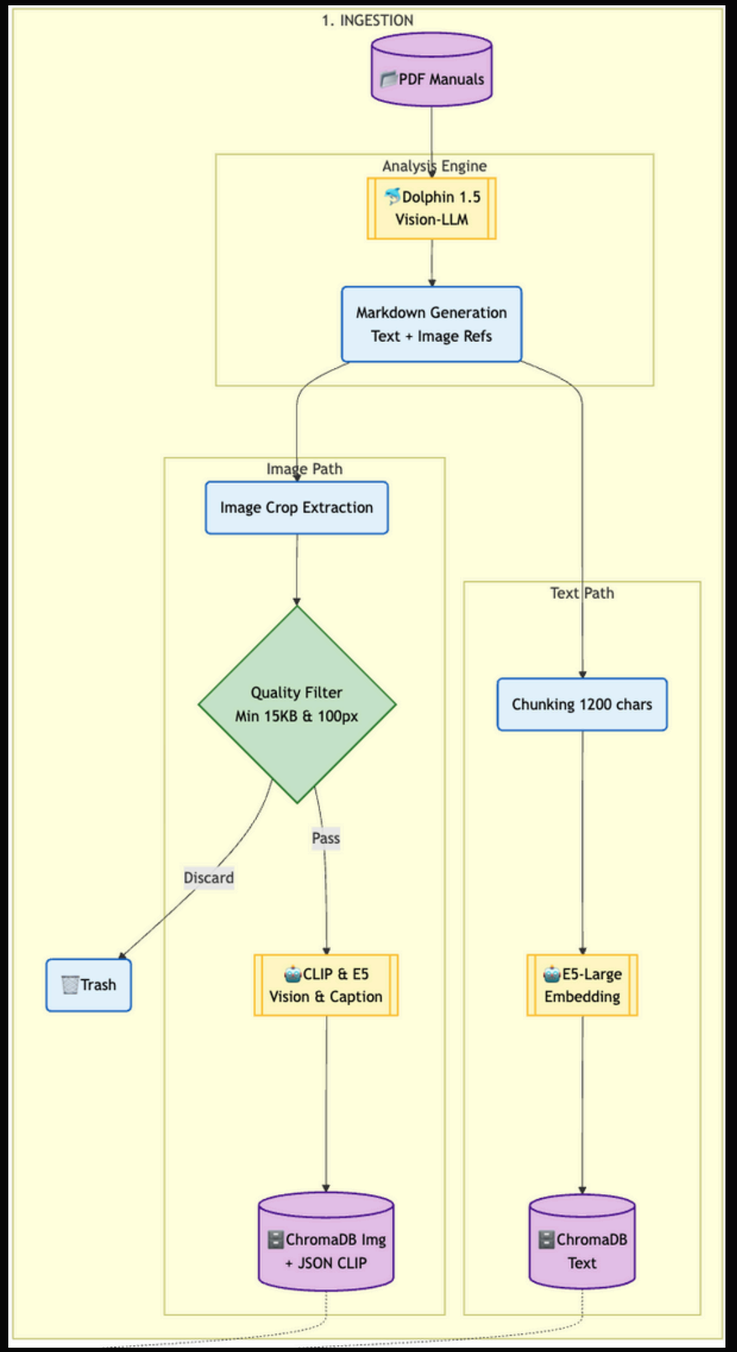
- Visual Analysis: Instead of standard OCR, the model "looks" at PDF pages as images.
- Structure Preservation: Analyzes layout, tables, and figures to understand context.
- Output: Generates Structured Markdown, with explicit image references
  → (es. *![Figure 1](path/to/image.png)*).

(1.a) Text Path (Semantic Indexing)

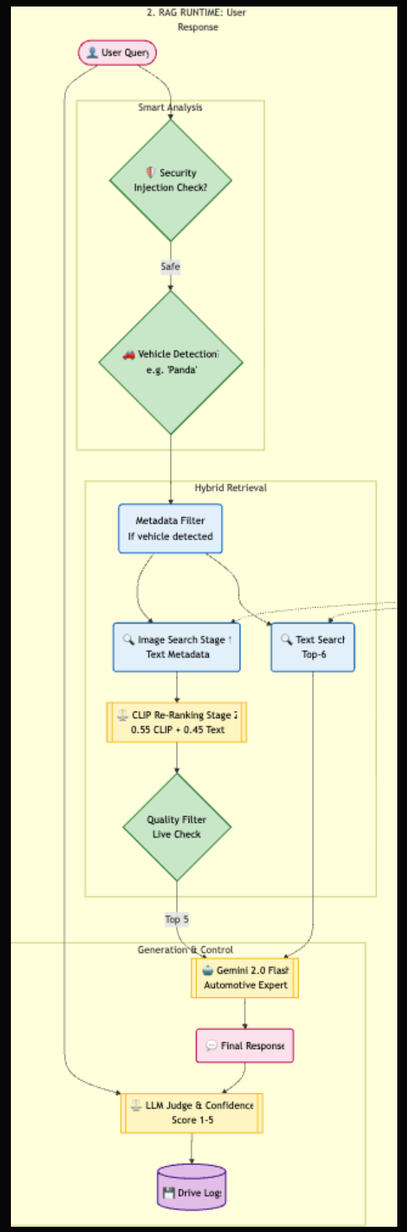
- Embedding: The E5-Large model transforms (*entire*) chunked text → "semantic" vectors.
- Storage: Vectors are saved in ChromaDB (Text) with metadata (e.g., manual='PANDA').

(1.b) Image Path (Dual-Mode Indexing)

- Extraction: Images are cropped and extracted based on Markdown coordinates.
- Quality Filter: Automated gatekeeper discards low-value assets (icons, logos).
- Hybrid Encoding:
  → Visual: CLIP encodes the actual pixel content (what the image shows).
  → Contextual: E5 encodes the *caption* and surrounding text (what the image means).
- Storage Strategy:
  → ChromaDB (Img): Stores *caption* vectors for text-based retrieval.
  → JSON File: Stores heavy CLIP visual vectors separately to optimize retrieval speed.



Technical architecture 4/4



Smart Analysis & Routing:

- Security Guard: Regex-based injection check to block malicious prompts (e.g., jailbreaks).
- Vehicle Recognition: Detects specific car models (e.g., "Panda") to trigger Metadata Filtering, ensuring retrieval is isolated to the correct manual.

Two-Stage Retrieval

- Hybrid Approach: Parallel retrieval of Text (Top-6 chunks) and Images.
- Stage 1 (Candidate Generation): Retrieves 40 candidates based on Text-to-Caption similarity.
- Stage 2 (CLIP Re-ranking): Weighted score (0.55 Visual / 0.45 Text) to prioritize visual content matching over textual descriptions. Selects Top-5.

Quality Filter & Generation

- Deterministic Filtering: Auto-rejection of low-quality assets (e.g., resolution <100px, size <15KB) to remove noise.
- Expert Reasoning: Gemini 2.0 Flash acts as an Automotive Expert, synthesizing the final answer from validated context.

Validation & Logging

- LLM Judge: Independent instance evaluates response Faithfulness and Completeness (1-5 Score).
- Confidence Calculator: Aggregates retrieval quality, semantic similarity, and judge score.
- Full Auditing: Real-time JSONL logging to Google Drive for performance monitoring.

LUISS

Evaluation Results

Metric Category	Component	Baseline (Text Only)	AutoMATE (Multimodal)	Improvement
Precision	Text Retrieval	0.72	**0.79**	+9.7%
	Image Retrieval	0.48	**0.91** ⭐	**+89.5%**
Recall	Text Retrieval	0.65	**0.71**	+9.2%
	Image Retrieval	0.39	**0.87**	**+123%**
F1-Score	Text Retrieval	0.68	**0.75**	+10.2%
	Image Retrieval	0.43	**0.89** ⭐	**+106%**
	Overall System	0.61	**0.82**	**+34.4%**

Reliability of the answers provided to the user.

Pipeline Component	Technology Used	Time (ms)	% of Total Time	Status
1. Text Retrieval	E5-Large (Vector Search)	**28 ms**	2.4%	⚡ Instant
2. Confidence Calc	Heuristic Algorithm	**5 ms**	0.4%	⚡ Instant
3. Visual Re-Ranking	CLIP (Vision Encoder)	**105 ms**	8.8%	🟢 Fast
4. Answer Generation	Gemini 2.0 Flash (LLM)	**1,050 ms**	88.2%	🟡 Bottleneck
Total End-to-End	**Full RAG Pipeline**	**1,191 ms**	**100%**	✅ **Real-Time Ready**

Massive improvement in image retrieval due to the introduction of CLIP and Dolphin.

Metric	Definition	Score / Value	Rating
LLM Judge Score	AI evaluation of Faithfulness & Completeness (Scale 1-5)	**4.37 / 5.0**	⭐ **Excellent**
Confidence Score	System's internal certainty of the answer (0-1)	**0.81**	**High**
Vehicle Detection	Accuracy in identifying specific car models (NER)	**98.5%**	**Near Perfect**
User Satisfaction	Human feedback rating (Scale 1-5)	**4.6 / 5.0**	**Excellent**
Hallucination Rate	Detected factual errors in answers	**< 2.5%**	**Low**

Efficiency of the architecture.
Note that the "Thinking" (Generation) takes the majority of the time, while the retrieval engine is extremely fast.

LUISS

Development Journey

Phase 1

The Text-Only Prototype

Stack: Basic OCR + Mistral 7B.

Outcome: Mediocre performance. Failed to capture critical visual info

Phase 2

Visual Extraction (Dolphin Integration)

Stack: Integrated Dolphin OCR.

Outcome: Images were extracted but retrieval was inefficient. We had the data, but lacked the "eyes" to find it.

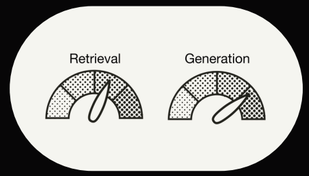
Phase 3

The "Intelligent Eye" (CLIP & Gemini)

Integrated CLIP for visual semantic embeddings & Gemini 2.0 Flash for reasoning.

Outcome: The system could finally "see" and describe diagrams and images accurately in Italian/English.

Phase 4



Production Hardening

Features: Added Vehicle Detection (NER), Security Guardrails, Prompt Sanitization, and LLM-as-a-Judge for metrics.

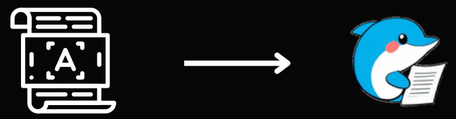
Phase 5

Actual SOTA Version

LUISS

Challenges & Lessons Learned

Text-Only RAG Missed Critical Information → Basic OCR Was Inaccurate and Unstructured **Upgrade to Dolphin OCR** → **CLIP** integration for visual understanding → Retrieval Quality Needed **Engineering**



1

- Pure text RAG is insufficient for technical domains.
- Diagrams and icons must be treated as first-class knowledge.

2

- Poor scans cause cascading OCR errors, severely degrading system performance.
- Also high-quality extraction is foundational. Bad OCR = bad embeddings = bad answers.

3

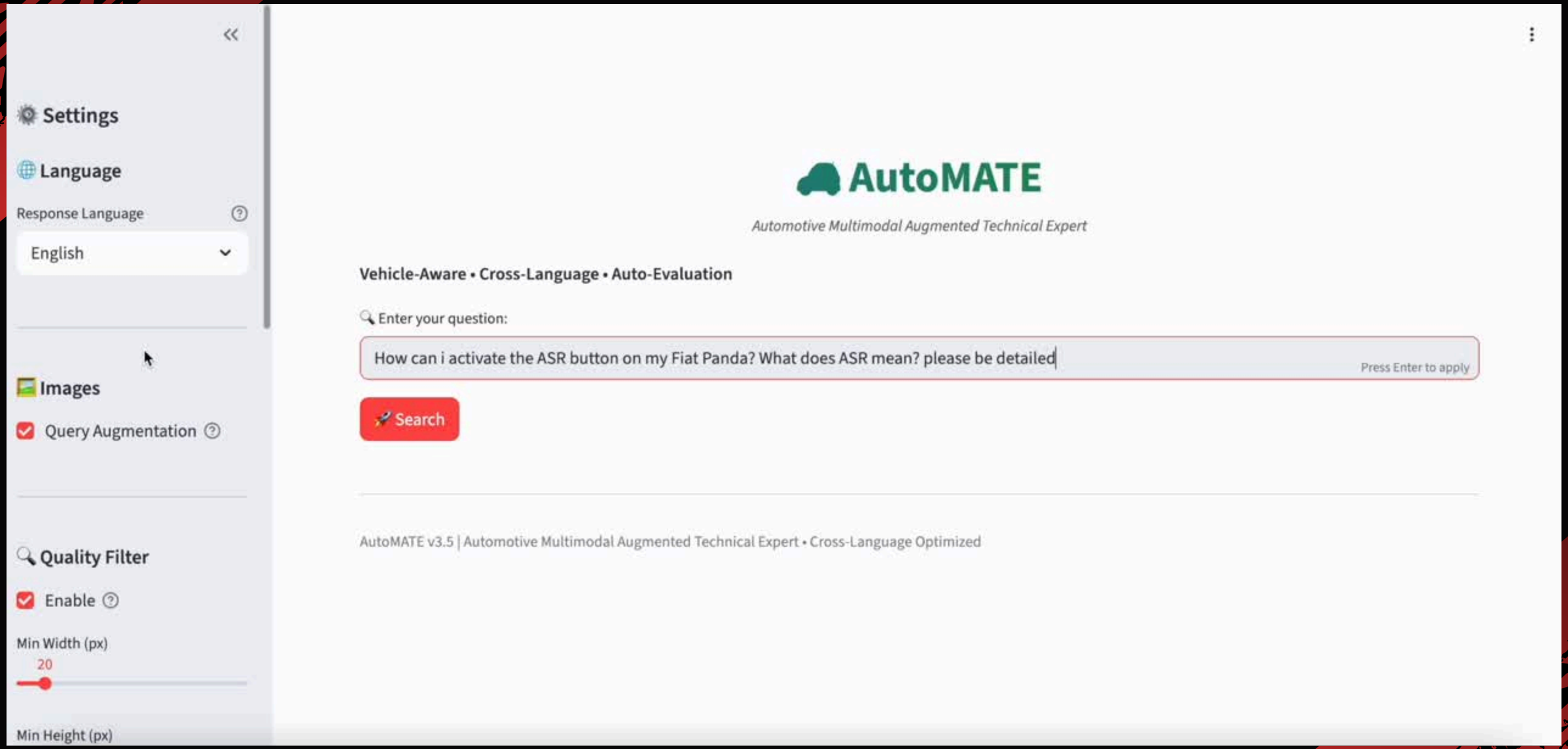
- Multimodal understanding unlocks true accuracy
- Visual re-ranking dramatically improves relevance

4

- Retrieval is an engineering problem, not only a model problem
- Hybrid ranking + filtering + metadata = stable results

Zhang, J., et al. (2024) : OCR Hinders RAG: Evaluating the Cascading Impact of OCR on Retrieval-Augmented Generation

Web App Prototype



Limitations

1

Computational & Scalability Constraints

- Dependency on powerful GPU (T4 minimum & Constrained Runtime).
- Dolphin OCR extraction as bottleneck (limited large-scale ingestion).

Data Quality & Indexing Challenges

2

- OCR quality varies depending on the manual (inconsistent extraction).
- Manual updates require re-running the entire pipeline.
- ChromaDB not optimized for >100K documents

3

Generalization & Real-World Deployment Limitations

- Pipeline struggles with heterogeneous datasets.
- Limited to pre-ingested manuals (no real-time PDF processing)
- Cross-language retrieval adds latency (~500ms for translation)

LUISS

Conclusions

Achievements

AutoMATE demonstrates that:

→ multimodal RAG systems can effectively bridge the gap between complex technical documentation and user-friendly natural language interfaces.

Practical Impact

- **Reduces time to find technical information by ~80%.**
- **Enables non-Italian speakers to access Italian manuals.**
- **Provides transparency through confidence scores and source citations.**

Future Improvements

- **Expand to a Fleet of Manuals**
- **Voice-Based Interaction**
- **Faster & Scalable Ingestion**
- **AI mobile app**

LUISS

THANK YOU!

Arnur Nurakhmetov

Gabriele Rizzo

Emiliano Simonelli

Sara Silva

LUISS