ViLeXa

Vietnamese Legal eXpert Assistant

Nguyen My Thong - 23521527
Tran Tuan Kiet - 23520822

CS311.Q11 - AI Programming Techniques
University of Information Technology - VNU-HCM

January 2026

The Problem: Why Legal AI is Hard

LLM Limitations

Hallucination — Legal advice MUST be verifiable

Knowledge Cut-off — Laws update frequently

Domain Gap — Vietnamese legal terminology

RAG as Solution

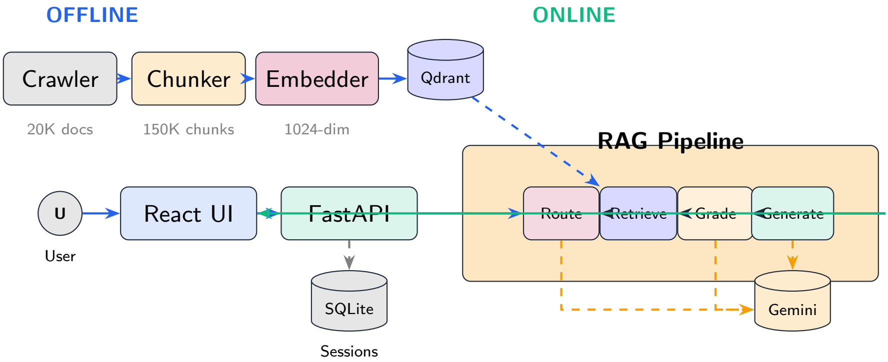
Grounded Answers — Cite specific Dieu, Khoan

Dynamic Updates — No retraining needed

Source Attribution — Traceable to documents

Our Goal: Build a Vietnamese legal QA system with **verifiable citations**

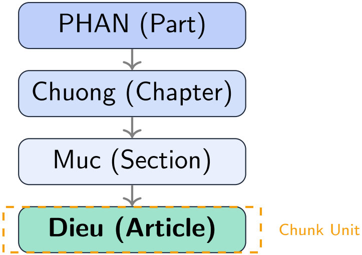
System Architecture: End-to-End Pipeline



Key Components: Vietnamese-Embedding-v2 — Qdrant Hybrid — LangGraph Workflow — Gemini 2.5 Flash

Data Preprocessing: Structure-Aware Chunking

Vietnamese Legal Hierarchy



Why Article-Level?

- ✓ Semantic completeness
- ✓ Natural legal boundaries
- ✓ Direct citation mapping

Chunk with Context Header

```
[Chuong I | Muc 1 | Dieu 15]
Dieu 15. Dieu kien kinh doanh
1. To chuc, ca nhan kinh doanh phai...
2. Truong hop kinh doanh nganh...
```

Preserved Metadata

`document_id`	Law identifier
`document_type`	Luat, Nghi dinh...
`phan, chuong, muc`	Hierarchy path
`dieu`	Article reference

20,410 docs → 150K chunks
Max 512 tokens — 50 token overlap

RAG Architecture: Design Decisions

Every component has multiple options - we evaluate each systematically

Slide 6

Embedding Model →

- GTE-multilingual
- BGE-M3
- **Vietnamese-Emb-v2**

Slide 6

Retrieval Strategy →

- Dense only
- Sparse only
- Hybrid (D+S)

Slide 7

Reranker (Optional) →

- None
- bge-reranker
- gte-reranker

Slide 8-9

RAG Architecture

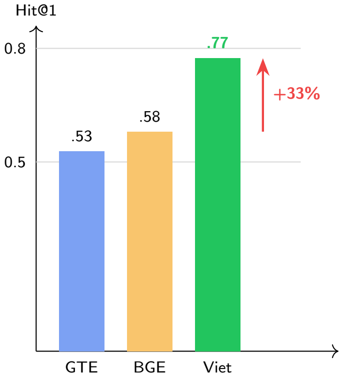
- Traditional RAG
- **Self-Reflective**

Methodology: Each decision backed by quantitative evaluation
150 queries — Zalo AI Legal Benchmark — LLM-as-Judge

Embedding Model Selection: Why Vietnamese-Specific?

Hypothesis: Domain-specific embedding outperforms multilingual for Vietnamese legal text

Model	Hit Rate			F1 Score		
	@1	@5	@10	@1	@5	@10
GTE Dense	.513	.827	.873	.513	.276	.159
GTE Sparse	.440	.720	.800	.440	.240	.146
GTE Hybrid	.527	.793	.880	.527	.264	.160
BGE-M3 Dense	.547	.847	.867	.547	.282	.158
BGE-M3 Sparse	.493	.793	.873	.493	.264	.159
BGE-M3 Hybrid	.580	.820	.900	.580	.273	.164
Viet-Emb Dense	**.773**	**.947**	**.953**	**.773**	**.315**	**.173**

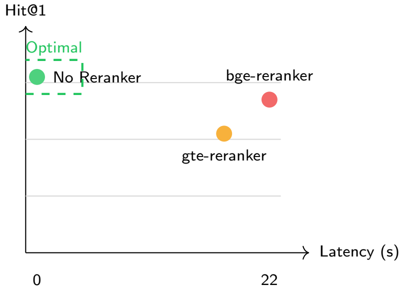


Winner: Vietnamese-Embedding-v2
77.3% vs 58.0% (BGE Hybrid)

Reranker Evaluation: Is It Worth the Cost?

Question: Does cross-encoder reranking improve retrieval quality?

Configuration	Hit@1	Hit@3	Hit@5	Latency
Dense Only	**0.773**	**0.900**	**0.947**	**0.13s**
+ bge-reranker-v2-m3	0.673	0.853	0.940	22.44s
+ gte-multi-reranker	0.520	0.867	0.913	8.36s



Surprising Finding:

- × Rerankers **decrease** Hit@1 by 10-25%
- × Latency increases **170x** (0.13s → 22s)
- ✓ Vietnamese-Embedding already well-calibrated

Decision: Skip reranker
High-quality embedding sufficient

Interpretation: General-purpose rerankers don't transfer well to Vietnamese legal domain

RAG Challenges: Why Self-Reflective?

Traditional RAG Failure Modes:

Missing Content

No relevant docs → hallucination

Missed Top-Ranked

Relevant doc ranked too low

Wrong Context

Non-legal query gets retrieval

Document Grading

Filter irrelevant before generation

Query Rewriting

Reformulate and retry (max 3x)

Query Routing

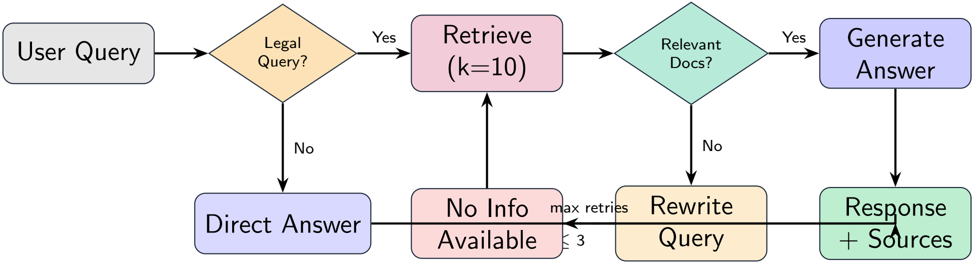
Skip retrieval for chit-chat

Traditional Query → Retrieve → Generate

Self-Reflective Query → Route → Retrieve → Grade → Generate

rewrite

Self-Reflective RAG: LangGraph Implementation



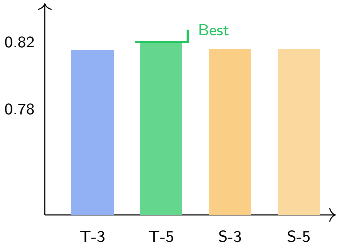
Key Features: Query intent classification — Relevance grading — Adaptive retry — Graceful failure

RAG Evaluation: Traditional vs Self-Reflective

Generation Quality Comparison

Config	Time	Tokens	Rel.	Faith.
Trad. k=3	1.81s	1550	0.781	0.904
Trad. k=5	**3.05s**	2472	**0.818**	**0.913**
Self-R k=3	6.62s	4069	0.784	0.912
Self-R k=5	8.06s	5736	0.785	0.902

Relevancy Score



Key Findings:

- ✓ Traditional k=5 achieves **best quality**
- ! Self-Reflective uses **3-4x more tokens**
- → Larger context (k=5) improves relevancy

Recommendation:
Traditional RAG (k=5)
for most queries

Detailed Quality Analysis: Pass Rates

Pass Rate at Different Thresholds

Config	Relevancy Pass % @0.5	@0.7	@0.9
Self-R k=3	84.2	69.9	56.2
Self-R k=5	82.9	71.2	56.2
Trad. k=3	82.9	74.0	56.2
Trad. k=5	**86.3**	**76.7**	**58.9**

Config	Faithfulness Pass % @0.5	@0.7	@0.9
Self-R k=3	93.8	90.4	80.8
Self-R k=5	94.5	91.1	76.7
Trad. k=3	**95.9**	89.0	75.3
Trad. k=5	95.9	88.4	77.4

Interpretation

86% relevancy $\geq$ 0.5

96% faithfulness $\geq$ 0.5

Faithfulness $>$ Relevancy

High Faithfulness (96%)
$=$ Low Hallucination Risk

Critical for legal domain

Evaluation Method:
LLM-as-Judge (Gemini)
DeepEval framework

Evaluation Methodology

Retrieval Metrics

Hit Rate @k — $\geq$1 relevant in top-k

Precision @k — Relevant / Retrieved

F1 Score @k — Harmonic mean

Generation Metrics

Relevancy — Addresses query?

Faithfulness — Grounded in context?

Evaluation Setup

Dataset: Zalo AI Legal Benchmark
Queries: 150 legal questions
Types: Factual, procedural, comparative
Judge: Gemini (LLM-as-Judge)
Framework: DeepEval

Experiment Coverage

7 Embedding | 4 Reranker | 4 RAG

= 2,250 evaluated retrievals

Technology Stack

Frontend: React + TypeScript + TailwindCSS

Backend: FastAPI + SQLAlchemy + JWT Authentication

RAG: LangChain + LangGraph + Gemini 2.5 Flash Lite

Vector DB: Qdrant + Vietnamese-Embedding-v2 (1024-dim)

Infrastructure: Docker Compose + NVIDIA GPU

Key Design Decisions

- ✓ Qdrant: native hybrid search
- ✓ LangGraph: stateful workflows
- ✓ Local GPU: fast embedding

Performance

- ✓ Retrieval: 0.13s per query
- ✓ Generation: 3.05s (k=5)
- ✓ 150K chunks indexed

Summary: Evidence-Based Decisions

Embedding Model

Vietnamese-Embedding-v2

+33% Hit@1 vs BGE-M3 Hybrid

Retrieval Strategy

Dense Only (no reranker)

Reranker hurts Hit@1 by 10-25%

RAG Architecture

Traditional RAG (k=5)

0.818 Relevancy — 0.913 Faithfulness

Self-Reflective RAG

For edge cases

Query rewriting for ambiguous queries

Best Configuration: Vietnamese-Embedding-v2 + Dense Retrieval + Traditional RAG (k=5)

Hit@1: 77.3% — Relevancy: 0.818 — Faithfulness: 0.913 — Latency: 3.05s

Limitations & Future Directions

Current Limitations

- Evaluation set: 150 queries
- No Vietnamese legal LLM
- Single-hop reasoning only
- No temporal awareness

Future Directions

- Expand benchmark
- Fine-tune legal embedding
- Multi-hop reasoning
- Temporal document filtering

Key Insight: Language-specific embedding yields greater returns than complex agentic workflows for Vietnamese legal domain

Conclusion

ViLeXa - Vietnamese Legal eXpert Assistant

Technical Contributions

1. Hierarchical chunking preserving legal structure
2. Systematic embedding evaluation (+33% improvement)
3. Self-reflective RAG with LangGraph

Practical Outcomes

1. End-to-end legal QA system
2. Evidence-based configuration
3. 96% low-hallucination responses

Hit@1	Relevancy	Faithfulness	Latency
77.3%	0.818	0.913	3.05s

Thank You!

Questions?

Nguyen My Thong 23521527@gm.uit.edu.vn
Tran Tuan Kiet 23520822@gm.uit.edu.vn

GitHub: github.com/trtkiet/ViLeXa