Comparative Evaluation of Agent-Based Financial Question Answering Systems Using RAG and LLMs

Mustafa Eren DALGIÇ - 2022556018

1. Introduction

Large Language Models (LLMs) are increasingly used in financial analysis systems for tasks such as bond evaluation, tax interpretation, policy reasoning, and investment decision support. However, pure LLM-based approaches often suffer from hallucination, lack of document grounding, and inconsistent reasoning when dealing with domain-specific financial data.

This project aims to design, implement, and evaluate **agent-based financial question answering systems** that combine:

- Retrieval-Augmented Generation (RAG)
- External financial tools
- Deterministic calculations
- Strict fallback and hallucination control

We conduct a **comparative benchmark study** between multiple LLM-based agents, focusing on **answer quality, reasoning reliability, hallucination resistance, and cost efficiency**.

---

2. Project Objectives

The primary objectives of this project are:

1. To design a **robust financial agent architecture** capable of:
    - Multi-hop reasoning
    - Numerical calculations
    - Document-grounded answers
    - Safe fallback behavior
2. To benchmark different LLM backends under identical conditions:
    - Groq (LLaMA-3 family)
    - GPT-4o-mini (OpenAI)
3. To evaluate trade-offs between:
    - Accuracy vs coverage
    - Cost vs reliability
    - Speed vs compliance-readiness
4. To provide **practical recommendations** for deploying financial agents in real-world systems.

---

3. System Architecture Overview

3.1 High-Level Architecture

Each system follows the same conceptual pipeline:

1. **Persistent Knowledge Base**
    - Financial documents (Eurobond bulletins, HSBC documentation, tax rules)
    - Indexed into a **persistent ChromaDB vector store**
2. **Retrieval-Augmented Generation (RAG)**
    - Semantic retrieval using sentence-level embeddings
    - Context injected directly into the LLM prompt
3. **Tool Layer**
    - Deterministic calculator for all numerical operations
    - External market data via yfinance
    - Explicit “NOT_FOUND” handling
4. **LLM Reasoning Core**

    - Groq or GPT-4o-mini
    - Zero-temperature inference for determinism
5. **Benchmark Orchestrator**
    - Scenario-based evaluation
    - JSON-based output logging
    - Error and fallback tracking

---

4. Knowledge Base Construction

4.1 Data Sources

The following documents were ingested into the ChromaDB knowledge base:

- Eurobond bulletins (daily transaction tables)
- HSBC Eurobond risk and policy documentation
- Turkish tax regulations for Eurobond income
- Fund prospectuses (AKE fund)
- Supplementary rate and policy files

4.2 Indexing Strategy

- Documents were chunked at sentence level
- Decimal numbers were preserved (no regex-based numeric corruption)
- Each chunk was stored with source metadata
- ChromaDB was configured in **persistent mode**, enabling reuse across sessions

This ensured **deterministic retrieval behavior** during benchmarking.

---

5. Benchmark Design

5.1 Scenario Categories

The benchmark consists of **over 40 scenarios**, grouped into the following categories:

1. **Comparison & Ranking**
    - Yield spreads
    - CDS comparisons
    - Bond ranking by yield or maturity
2. **Logic & Math**
    - Coupon income calculations
    - Threshold exceedance
    - Percentage differences
3. **Multi-Hop Reasoning**
    - Cross-document inference
    - Date-based settlement logic
    - Policy + numerical chaining
4. **Fallback Behavior**
    - Missing bond data
    - Conditional secondary queries
5. **Hallucination Resistance**
    - Explicitly forbidden price invention
    - Non-existent concepts
6. **Temporal Reasoning**
    - Yield trends over time
    - Duration calculations
7. **Policy & Risk Interpretation**

    - Liquidity risk
    - Withdrawal rights
    - Investor eligibility
8. **Complex Scenarios**
    - Multi-constraint filtering
    - Mixed quantitative and qualitative reasoning

---

6. Agent Implementations

6.1 Groq-Based Agent

- Model: LLaMA-3 (via Groq API)
- Strengths:
    - Extremely low latency
    - Very low cost per token
    - High answer completion rate
- Limitations:
    - Occasionally infers missing information
    - Requires strong external guardrails

6.2 GPT-4o-mini–Based Agent

- Model: GPT-4o-mini
- Strengths:
    - Strong document grounding
    - Conservative, compliance-oriented reasoning
    - Excellent hallucination resistance
- Limitations:
    - Higher cost per token
    - Slightly lower answer coverage

---

7. Evaluation Results

7.1 Answer Quality

- **GPT-4o-mini** consistently produced well-structured, source-aligned answers
- **Groq** answered more questions overall but sometimes relied on implicit assumptions

7.2 Multi-Hop Reasoning

- GPT-4o-mini handled cross-document logic more reliably
- Groq occasionally terminated early or skipped intermediate reasoning

7.3 Hallucination and Safety

- GPT-4o-mini explicitly stated missing data when appropriate
- Groq occasionally produced partial or inferred answers under ambiguity

7.4 Policy Interpretation

- GPT-4o-mini used more formal financial language
- Groq responses were shorter and less explicit

---

8. Cost Analysis (Token-Based)

8.1 Pricing Assumptions (Approximate)

Model	Input Cost	Output Cost
Groq (LLaMA-3)	~$0.0001 / 1K	~$0.0001 / 1K

Model	Input Cost	Output Cost
GPT-4o-mini	~$0.15 / 1M	~$0.60 / 1M

8.2 Estimated Cost per Scenario

Model	Cost per Scenario
Groq	~$0.00035
GPT-4o-mini	~$0.0011

8.3 Full Benchmark Cost (~40 scenarios)

Model	Total Cost
Groq	~$0.014
GPT-4o-mini	~$0.044

---

9. Comparative Summary

Dimension	Groq	GPT-4o-mini
Cost Efficiency	Very High	Medium
Answer Reliability	Medium	High
Hallucination Resistance	Medium	Very High
Compliance Suitability	Medium	High
Production Readiness (Finance)	Medium	Very High

---

10. Key Insight

Failure to answer due to missing data is not a weakness in financial agents.
In regulated environments, *refusal is often the correct and safer behavior*.

Thus, a system that answers fewer questions but avoids hallucination may be **superior** to one that answers more questions inaccurately.

---

11. Final Conclusion

This project demonstrates that:

- **Groq-based agents** are ideal for:
  - Large-scale experimentation
  - Cost-sensitive workloads
  - Exploratory analysis
- **GPT-4o-mini–based agents** are better suited for:
  - Client-facing financial applications
  - Compliance-sensitive environments
  - High-stakes decision support

Recommended Deployment Strategy

A **hybrid architecture** is recommended:

- Use Groq for bulk reasoning and pre-analysis
- Use GPT-4o-mini for final validation and delivery

---

12. Future Work

Future improvements may include:

- Automated confidence scoring
- Ensemble agent voting

- Rule-based post-validation layers
- Fine-tuned financial-domain LLMs

Comparative Evaluation of Agent Outputs and Cost Efficiency

1. Scope of Comparison

This evaluation compares two agent-based systems tested on the same **scenario-driven benchmark** consisting of comparison, logic & math, multi-hop reasoning, temporal reasoning, hallucination resistance, and policy interpretation tasks.

The systems compared are:

- **Groq-based Agent** (LLaMA-3 family via Groq API)
- **GPT-4o-mini–based Agent** (OpenAI API)

Both agents were tested using:

- The same question set
- The same document corpus (Eurobond bulletins, HSBC documentation, tax summaries)
- Similar RAG-style context injection

The outputs were normalized into JSON format before analysis.

---

2. Answer Quality and Reasoning Accuracy

2.1 Quantitative & Logical Tasks (Math, Spreads, Percentages)

Observation

- Both agents perform strongly on deterministic math tasks (basis point spreads, percentage differences, coupon calculations).
- Groq occasionally produced **numerical answers even when the supporting context was incomplete**.
- GPT-4o-mini was more conservative, frequently stating *"information missing"* when explicit data was not found.

Assessment

- **Groq**: Higher task completion rate, but higher risk of *implicit assumption*.
- **GPT-4o-mini**: Lower hallucination risk, stricter adherence to "use only context".

Winner:

- *Accuracy-first evaluation*: **GPT-4o-mini**
- *Completion-first evaluation*: **Groq**

2.2 Multi-Hop & Cross-Document Reasoning

Observation

- GPT-4o-mini consistently traced answers back to explicit document evidence (e.g., T+2 settlement logic, tax threshold conversions).
- Groq sometimes returned partial reasoning or stopped early when documents were fragmented.

Assessment

- GPT-4o-mini shows stronger **document grounding** and **chain-of-evidence discipline**.
- Groq is faster but less consistent when multiple documents must be combined.

Winner: **GPT-4o-mini**

2.3 Hallucination & Fallback Behavior

This category is critical for financial agents.

Observation

- GPT-4o-mini reliably:

    - Refused to invent missing prices
    - Explicitly stated when data was unavailable
    - Correctly handled fallback logic (e.g., "If missing, state it and give AAPL price")
- Groq:
    - In some cases provided answers marked as missing_or_incomplete = true
    - Occasionally inferred information instead of strictly refusing

Assessment

- GPT-4o-mini is significantly more robust against hallucination.
- Groq is usable but requires **stronger system-level guardrails**.

Winner: **GPT-4o-mini**

2.4 Policy & Risk Interpretation (Narrative Tasks)

Observation

- Both models performed well on descriptive policy questions (liquidity risk, emergency cash, withdrawal rights).
- GPT-4o-mini responses were more structured and aligned with formal financial language.
- Groq responses were shorter and sometimes less explicit.

Winner: **GPT-4o-mini**

---

3. Coverage vs Precision Trade-off

Dimension	Groq Agent	GPT-4o-mini Agent
Answer Coverage	Higher	Moderate
Precision	Moderate	High
Hallucination Risk	Medium	Low
Missing Data Handling	Weaker	Strong
Financial Compliance Suitability	Medium	High

Interpretation
If an agent *cannot solve a question due to missing data*, that does **not automatically mean the agent is bad**. In regulated or financial contexts, **refusal is often the correct behavior**.

---

4. Cost Comparison (Token-Based)

4.1 Pricing Assumptions (Approximate, 2025)

Model	Input Tokens	Output Tokens
Groq (LLaMA-3-70B)	~$0.0001 / 1K	~$0.0001 / 1K
GPT-4o-mini	~$0.15 / 1M	~$0.60 / 1M

Note: Groq pricing is dramatically cheaper due to hardware acceleration and aggressive subsidization.

4.2 Estimated Cost per Scenario

Assuming per scenario:

- ~3,000 input tokens
- ~500 output tokens

Model	Cost per Scenario
Groq	~$0.00035
GPT-4o-mini	~$0.0011

4.3 Estimated Cost for Full Benchmark (~40 Scenarios)

Model	Total Cost
Groq	~$0.014
GPT-4o-mini	~$0.044

---

5. Cost-to-Quality Ratio

Criterion	Groq	GPT-4o-mini
Cost Efficiency	⭐ ⭐ ⭐ ⭐ ⭐	⭐ ⭐ ⭐
Reliability	⭐ ⭐ ⭐	⭐ ⭐ ⭐ ⭐ ⭐
Compliance-Readiness	⭐ ⭐ ⭐	⭐ ⭐ ⭐ ⭐ ⭐
Production Safety	⭐ ⭐ ⭐	⭐ ⭐ ⭐ ⭐ ⭐