Technical Report

Causal Retrieval-Augmented System for Conversational Analysis

---

1. Introduction

Customer support systems generate large volumes of conversational data, but existing retrieval and analytics approaches primarily focus on **semantic similarity** rather than **causal explanation**. This limits their usefulness for analytical questions such as *why an escalation occurred*, *what triggered an investigation*, or *what factors led to refunds or fraud actions*.

This project proposes a **Causal Retrieval-Augmented Generation (Causal RAG)** system that retrieves relevant conversations, extracts interpretable causal signals from dialogue turns, and produces **transparent, evidence-backed explanations** rather than free-form text.

---

2. Problem Statement

Given:

- A dataset of customer service conversations
- A set of analytical and operational queries

The system must:

1. Retrieve relevant conversations and dialogue turns
2. Identify causal factors contributing to an outcome
3. Provide supporting evidence from transcripts
4. Support follow-up analytical questions
5. Output results in a structured CSV format

The goal is **faithfulness, explainability, and auditability**, not generative fluency.

---

3. Dataset Description

3.1 Conversational Dataset

The system uses a conversational transcript dataset where each record contains:

- transcript_id
- domain and intent metadata
- multi-turn conversations between customer and agent

Each conversation is flattened into **turn-level records** for fine-grained retrieval and analysis.

3.2 Query Dataset

A curated dataset of **50 queries** is constructed to ensure:

- Coverage across multiple domains
- Sufficient complexity
- Support for follow-up reasoning

Query Categories

- Delivery Issues
- Refunds
- Fraud
- Security
- Account Updates
- Product Issues
- Payment Problems
- Multi-Issue Scenarios
- Causal / Analytical Queries

Each query entry contains:

- Query_Id
- Query
- Query_Category

The system processes this dataset in batch mode and produces a submission-ready output CSV.

---

4. System Architecture

The system follows a **two-stage semantic retrieval pipeline combined with causal aggregation**, designed to balance retrieval accuracy with computational efficiency.

4.1 Conversation-Level Retrieval

All dialogue turns are embedded using the **Sentence-BERT model all-MiniLM-L6-v2**, which produces fixed-length dense semantic vectors.
Turn embeddings belonging to the same transcript are **mean-pooled** to construct a single **conversation-level embedding** representing the overall semantic context.

Incoming queries are embedded using the same transformer model and compared against conversation vectors using **cosine similarity**.

The **top-K most similar conversations** are selected as candidates for fine-grained evidence extraction.

4.2 Turn-Level Evidence Retrieval

From the shortlisted conversations, individual dialogue turns are retrieved along with their precomputed embeddings.
Each turn embedding is compared to the query embedding using **cosine similarity**, enabling fine-grained semantic ranking at the turn level.

The **top-K highest-scoring turns** are selected as candidate evidence for causal analysis.
This hierarchical retrieval strategy reduces noise, improves evidence relevance, and avoids exhaustive turn-level search across the entire dataset.

---

5. Causal Tagging and Feature Extraction

Each retrieved dialogue turn is passed through a **deterministic, rule-based causal tagger**.

Extracted Causal Signals

- Customer frustration (none / high)
- Repetition of issues (yes / no)
- Escalation signals (none / weak / strong)
- Agent action (apology / explanation / resolution / none)
- Policy reference (yes / no)

This design ensures:

- Transparency
- Reproducibility
- No hallucinated causal claims

The system is intentionally designed so this module can later be replaced by an LLM-based annotator if needed.

---

6. Causal Aggregation Logic

Causal tags from all evidence turns are aggregated using frequency-based counters.

Dominant causal factors are identified using **explicit thresholds** (e.g., ≥40% of evidence turns), ensuring:

- Clear decision logic
- Deterministic behavior
- Easy auditability

Example dominant factors:

- High customer frustration
- Repeated unresolved issues
- Explicit escalation signals
- Delayed or insufficient resolution

---

7. Explainable Output Generation

For each query, the system outputs:

- Inferred outcome type
- Dominant causal factors
- Supporting evidence with:
  - transcript ID
  - turn ID
  - speaker
  - original text
  - causal tags

All outputs are **grounded in retrieved evidence** — no unsupported reasoning is generated.

---

8. Follow-up Query Handling

A session-level memory object stores:

- Active query
- Dominant causal factors
- Retrieved evidence
- Outcome type

This enables follow-up questions such as:

- *“Which factor mattered the most?”*
- *“Show evidence for that.”*
- *“Why did this happen?”*

The system reuses stored evidence instead of re-retrieving, ensuring consistency and faithfulness.

---

9. Evaluation Strategy and Results

9.1 Evaluation Criteria

The system is evaluated qualitatively on:

- Retrieval relevance
- Evidence faithfulness
- Causal interpretability
- Robustness to diverse query categories

9.2 Observations

- Two-stage retrieval significantly improves evidence relevance compared to turn-only retrieval.
- Rule-based causal tagging produces stable and interpretable results.
- The system consistently grounds explanations in actual transcript evidence.
- Multi-issue and causal queries are handled without requiring retraining.

9.3 Limitations

- Rule-based tagging may miss subtle linguistic cues.
- Outcome inference is keyword-based and can be extended.
- No probabilistic causal inference is performed.

---

10. Future Work

- Replace rule-based tagging with LLM-based structured annotation
- Introduce learned rerankers for turn selection
- Extend outcome inference using supervised classifiers
- Add causal graphs for inter-factor relationships

---

11. Conclusion

This project demonstrates that **causal explainability** can be integrated into retrieval systems without sacrificing transparency or reliability. By combining semantic retrieval with deterministic causal reasoning, the system provides faithful, auditable explanations suitable for real-world analytical use cases