Technical Report: Call Compliance and Quality Analysis System

1. Introduction--

This project focuses on analysing customer–agent conversations to evaluate compliance and professionalism. The system addresses three specific areas:

- **Question 1:** Detecting profanity used by either the agent or the customer.
- **Question 2:** Identifying whether an agent disclosed sensitive account information before verifying the customer's identity.
- **Question 3:** Measuring call quality using silence and overtalk percentages.

The inputs are JSON or YAML transcripts. Each transcript is structured into utterances with four fields: speaker, text, stime, and etime. Filenames act as call IDs.

The implementation includes two approaches for Q1 and Q2: a **regex-based pattern matching method** and an **LLM-based method** using Google Gemini. For Q3, the system calculates overtalk and silence metrics and provides both interactive visualizations in Streamlit and aggregate insights through a Jupyter notebook.

2. System Design--

Pattern Matching Approach

- **Normalization:** All text is normalized using NFKC and unidecode so that obfuscations like f@#k are captured.
- **Regex:** Predefined regexes identify curse words and customer identity verification (DOB, SSN, account numbers, etc.).
- **Logic:** Flags whether profanity was used and whether Sensitive information was shared before verification.

This approach is **fast, deterministic, and cost-effective**, but it only works if the vocabulary and patterns are known.

Machine Learning / LLM Approach

- **Prompt engineering:** A structured Gemini prompt defines profanity rules and compliance definitions.
- **Parsing:** Output is parsed into standardized CSV format.
- **Context handling:** Goes beyond regex by understanding order (e.g., whether verification happened before disclosure).

This approach is **flexible and context-aware** but depends on model capabilities, API calls, costs more, and runs slower.

Visualization and Metrics (Q3)

- **Intervals:** Speech segments from each speaker are converted into intervals.
- **Overtalk:** Computed as intersection of agent and customer speech intervals.
- **Silence:** Computed as total call time minus the union of all speech intervals.
- A dedicated second page allows either uploading a ZIP of calls or pasting a single JSON prompt for per-call visualization.
- A separate notebook provides aggregate analysis across all calls.

3. Implementation Recommendations--

Q1: Profanity Detection

Regex Method:

- Works well for standard curse words and obfuscations.
- Can be deployed in real time on large datasets.
- Limitation: cannot detect creative slang or implied insults.

LLM Method:

- Can capture indirect or nuanced profanity (e.g., sarcasm, coded language).
- Useful for auditing or when the language domain evolves quickly.
- Limitation: slower, more expensive, and occasionally inconsistent.

Recommendation: Use ***regex*** as the primary production method, due to its reliability and speed. Add an LLM audit layer for ambiguous cases or for periodic reviews.

Q2: Privacy and Compliance Violation

Regex Method

- Good at spotting raw PII tokens like dates of birth or account numbers.
- Limitation: cannot reliably judge whether disclosure happened *before* verification.

LLM Method

- Context-sensitive: understands whether verification was completed before disclosure.
- Can detect subtle violations (e.g., when an agent confirms a number the customer mentions).
- Limitation: requires prompt tuning and validation to avoid misclassification.

Recommendation: For compliance, ***the LLM*** method should be the default since context matters. Regex can still be valuable as a pre-filter for identifying potential PII mentions.

4. Visualization Analysis (Q3)--

I extended the system with a second Streamlit page dedicated to call quality metrics, alongside a Jupyter notebook for definite insights. This setup enables both **per-call review** and **aggregate reporting**.

Insights from Visualization

1. **Weighted vs. Average Percentages**
    - Overtalk hovers around 10% across the organization, whether measured by weighted or mean averages.
    - Silence averages about 3–4%, which is relatively low.
    - This shows that overtalk is a more consistent behavior than silence.
2. **Distribution of Overtalk**
    - Most calls cluster around 8–15% overtalk.
    - The mean (10.5%) and median (10.0%) align, confirming this is a regular feature of calls.
    - Outliers with over 30–40% overtalk exist and should be reviewed individually.
3. **Distribution of Silence**
    - Silence is minimal in most calls (median 1.3%).
    - Outliers show 8–12% silence, which may reflect agents hesitating, long holds, or customers being unresponsive.
4. **Call Duration vs. Overtalk**
    - Very short calls (<40s) sometimes have extremely high overtalk percentages.
    - Longer calls (60–100s) stabilize with overtalk in the 5–15% range.
    - This suggests rushed or chaotic starts but smoother conversations later.

5. Conclusion--

This project demonstrated how multiple approaches—**regex-based pattern matching** and **LLM-powered methods**, can be applied to the analysis of debt collection call transcripts.

Regex provides a transparent, rule-driven baseline, while the integration of LLMs adds further potential, as they can interpret context, semantics, and variations in natural language beyond the reach of simple rules or small classifiers or RegEx.

- Our experiments showed that while regex ensures precision in well-defined cases, **LLMs offer the promise of higher accuracy and richer contextual understanding**. However, their performance depends on model choice, training data, and deployment constraints (cost, latency, interpretability).
- Going forward, **exploring more diverse LLMs and their capabilities, ranging from small domain-specific models to larger state-of-the-art architectures, could lead to improved detection of profanity, PII disclosures, and compliance risks**. Fine-tuning LLMs on domain data or combining them with weak supervision strategies may further enhance both precision and recall.
- Additionally, hybrid pipelines, that is, using regex or ML for high-confidence matches and LLMs for context-rich cases, can provide a balanced approach.
- In summary, while this project successfully delivers the required profanity, compliance, and call quality analyses along with a deployed Streamlit application, this exploration could unlock more reliable, scalable, and context-aware solutions for real-world compliance monitoring in customer interactions.