**Reddit Insights Chatbot with RAG  
MSDS692 – Data Science Practicum 1**

**Jonish Bishwakarma**

**Progress Report for Week7**

**Project Details**This project aims to build a research tool that uses Reddit discussions to surface industry-specific insights, particularly about commonly used software and related pain points in law firms, construction, and tech. Posts and comments are scraped from selected subreddits, cleaned, and analyzed for software/tool mentions. A Retrieval-Augmented Generation (RAG) chatbot will later be developed so users can query insights conversationally. The project also compares RAG performance with an LLM-only baseline.

**Project Timeline:**

Week 1 – Project definition and submit proposal (DONE)

Week 2 – Data collection (DONE)

Week 3 – Data cleaning, preprocessing, and initial keyword filtering (DONE)

Week 4 – Build software dictionary and extract mentions (DONE)

Week 5 – Sentiment analysis and identification of pain points (DONE)

Week 6 – Develop and test RAG + LLM pipelines (DONE)

Week 7 – Evaluation of RAG vs LLM and refinement (DONE)

Week 8 – Final analysis, presentation preparation, and submission (Pending)

**Planned Work for the Week:**  
This week’s objectives were to evaluate and compare the RAG pipeline’s performance against the baseline LLM model using a fixed set of in-scope and out-of-scope questions. The goal was to analyze retrieval accuracy, contextual relevance, and the model’s ability to distinguish between relevant and irrelevant queries. Additional refinements were planned to improve retrieval performance and integrate evaluation results into the web interface.

**Progress for the Week:**

* Implemented a structured evaluation script (evaluate.py) to automatically test the RAG and LLM pipelines using 20+ mixed-scope questions.
* Added accuracy, precision, recall, and F1-score calculations to objectively assess each model’s performance.
* Integrated evaluation execution and results visualization directly into the Flask web app, enabling one-click evaluation and summary display.
* Observed realistic performance differences: RAG achieved 65% accuracy, 70% F1-score, while LLM scored 100% due to its general response coverage.
* Enhanced RAG retrieval quality through improved query filtering and increased document chunk overlaps in Pinecone indexing.
* Updated front-end interface (index.html) to display evaluation tables and average scores dynamically after evaluation runs.

**Roadblocks/Issues:**

* RAG still returns “I don’t know” for several in-scope queries due to limited context coverage from Reddit data.
* Indexing and retrieval performance depend heavily on embedding quality and chunking parameters; further optimization could improve recall.
* Occasional issues with Pinecone index initialization required manual index creation before running the Flask app.
* Evaluation outputs initially showed inflated (100%) scores due to overly simplified scoring logic, which was later fixed with contextual validation.

**Plan for Next Week:**

* Prepare final project documentation, visualizations, and presentation materials.
* Create evaluation summary visualizations comparing RAG vs LLM metrics.
* Add final usability polish to the chatbot interface (clean logs, progress bars, and evaluation summaries).
* Finalize the README and ensure all pipeline scripts execute seamlessly from the web interface.

**Resources for the Week:**

* LangChain, Pinecone, HuggingFace Sentence Transformers, and OpenAI GPT-3.5-turbo.
* Flask for backend and UI integration with server-sent events for live log updates.
* Existing Reddit dataset indexed in Pinecone for contextual retrieval