Project Proposal – CS7290

Sashank Vasepalli

Introduction:

Large Language Models are taking the world by storm. They have crept into every industry, looking to give a significant technological lead to its early adaptors. The most popular and in-demand jobs since Q2 2023 are in Generative AI and LLMs. However, one of the industries that is slow to adapt is Finance. This stems from the combination of the risk aversion nature of the financial sector, insufficient infrastructure, data privacy concerns, and a lack of interpretability.

Objective:

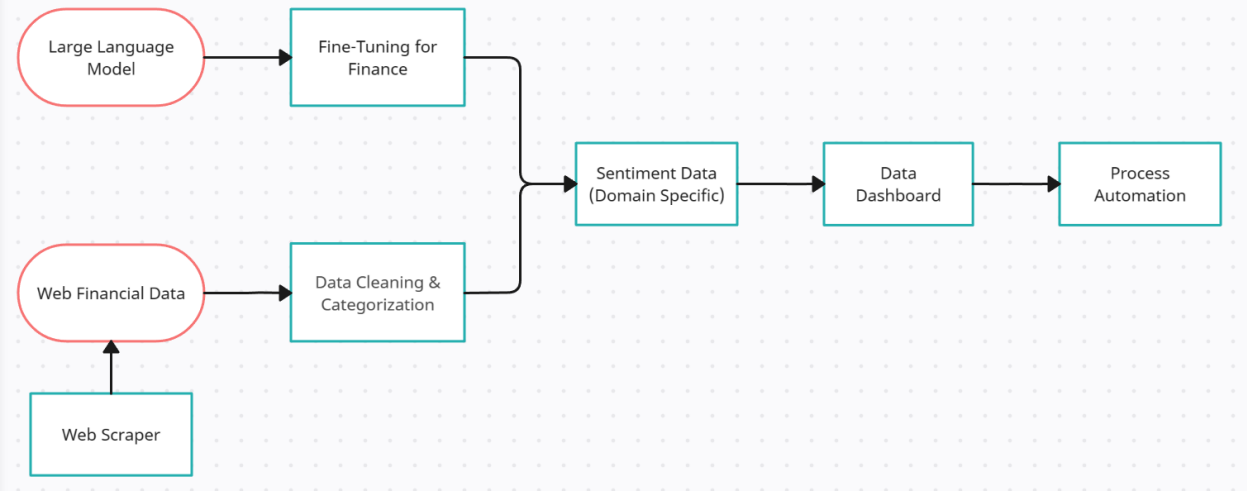
My objective for this project is to develop a system that performs sentiment analysis on financial data that is readily available on the internet, using LLMs. Due to the lack of high quality, open-source fine-tuned LLMs for finance [1)], I aim to build a tool to provide up-to-date sentiment data for different financial domains.

Scope:

The objective of my system is to use a LLM fine-tuned for financial data [2)] to perform sentiment analysis for different financial domains across varying time frames: Short (10 Day), Medium (50 Day), Long (200 Day). I plan to modify an open-source financial web scraper [3)] to obtain live financial data, clean and extract useful metadata, use an LLM for sentiment analysis, and provide an online dashboard. As a stretch objective, I plan to automate the process and host it using leading cloud services.

I do NOT plan on developing a trading algorithm using sentiment, as trading algorithms by themselves are extremely complex, and high-quality sentiment data would only serve to be a small part of the ensemble trading system [4)].

Methodology:



1. Select the AdaptLLM 13B model [2)], which is already fine-tuned for financial data.
2. Fine-tune the LLM using Prompt Engineering [6)], and test with 2 datasets [7)]: FinancialPhraseBank and FiQA.
3. Identify widely used financial documents on the web, ideally with APIs or readily available web scrapers [3)] (Ex. Reddit-WallStreetBets, Wall Street Journal, Twitter, Yahoo Finance).
4. Modify the readily available web scraper [3)] for scraping and storing necessary data.
5. Clean incoming scraped data, extract metadata (views, outreach), and categorize into the following different domains using keywords: Equity Markets, Fixed Income, Commodities, Forex, Real Estate, Private Equity, Cryptocurrency.
6. Feed data to the LLM using the newly tested prompt, to obtain sentiment data and confidence scores.
7. Devise an algorithm (Moving Average, EMA) to add sentiment data and confidence scores, to generate sentiment across varying time frames: Short (10 Day), Medium (50 Day), Long (200 Day).
8. Create an easily digestible data dashboard to display the above data.
9. Stretch: Automate processes and host on online using cloud services.

Timeline:

1. February 6 – Setup environment and ensure working of the AdaptLLM.
2. February 13 – Prompt Engineering tests using one of the test datasets.
3. February 20 – Identify and set up a web scraper for one of the data sources.
4. February 27 – Run the web scraper and set up a pipeline for cleaning data.
5. March 5 – Develop the categorization algorithm using bag of words.
6. March 12 – PPT to demonstrate LLM tests, web scraper, and categorization pipeline.
7. March 19 – Use LLM for inference on categorized data.
8. March 26 – Algorithm and Dashboard for sentiment data display.
9. April 2 – Write report and paper for project.
10. April 9 – Spare week for unfinished work. Expand to more data sources if feasible.
11. April 16 – PPT preparation and presentation.

References:

1) Large Language Models in Finance: A Survey (arxiv.org)

2) AdaptLLM/finance-LLM-13B · Hugging Face

3) je-suis-tm/web-scraping: Detailed web scraping tutorials for dummies

4) Sentiment Trading and Hedge Fund Returns - CHEN - 2021 - The Journal of Finance - Wiley Online Library

5) meta-llama/Llama-2-7b-hf · Hugging Face

6) oobabooga/text-generation-webui: A Gradio web UI for Large Language Models. Supports transformers, GPTQ, AWQ, EXL2, llama.cpp (GGUF), Llama models. (github.com)

7) Financial Sentiment Analysis (kaggle.com)