High Level Design Document: Telegram RFQ Bot

1. Introduction

The purpose of this document is to outline a high-level design of a Telegram bot that responds to Requests For Quotations (RFQs). The system aims to reduce the time spent by traders in answering multiple RFQs by automating this process using the GPT-4 large language model (LLM).

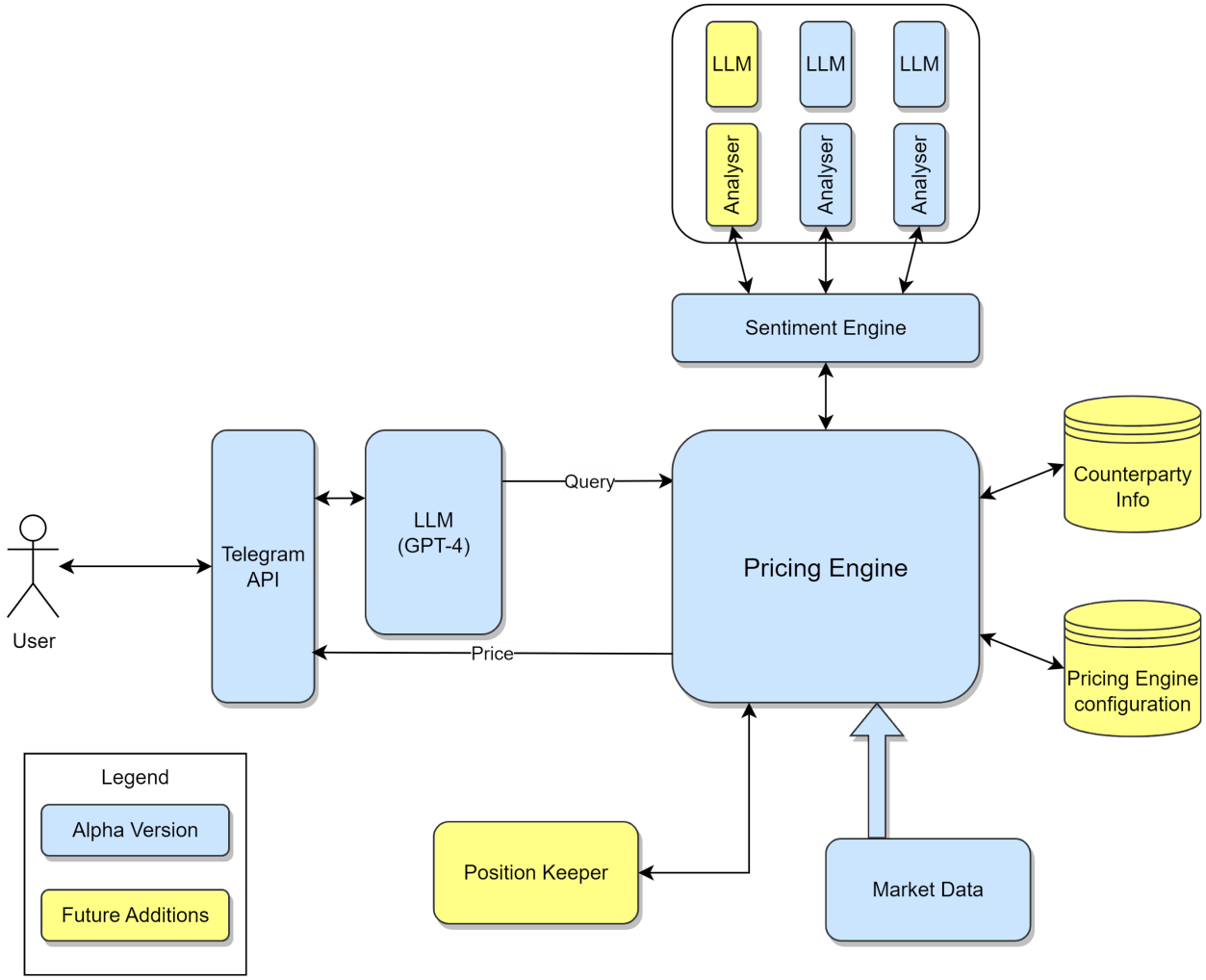
2. System Overview

The application integrates with the Telegram API to receive messages, which are subsequently processed by the GPT-4 LLM. If the GPT-4 LLM cannot understand a request, it responds accordingly. In case the request can be processed, the GPT-4 LLM produces a query in JSON format, which is then sent to the pricing engine.

The pricing engine, in turn, integrates with a market data feed for pricing the instrument and a sentiment analysis engine for skewing prices. The sentiment analysis engine scrapes news from multiple sources and uses the GPT-3.5 LLM to infer the sentiment from the latest news.

Most of the pricing engine configurations are currently hardcoded, but can be moved to a database in future iterations. Similarly, counterparty information could be stored in a database in future designs.

3. System Architecture



3.1 Telegram Bot

The bot integrates with the Telegram API and is the primary interface for users to submit RFQs.

3.2 GPT-4 LLM

This module processes incoming messages and formulates queries based on the messages. It has a fall-back mechanism to communicate its inability to understand a message when it can't be parsed into a suitable query.

3.3 Pricing Engine

The Pricing Engine receives queries from the GPT-4 LLM and calculates prices. It uses market data feeds and sentiment analysis results to provide the best prices. Prices are also skewed based on the counterparty, market volatility, and current position.

3.4 Sentiment Analysis Engine

The Sentiment Analysis Engine employs GPT-3.5 LLM to infer the sentiment from the latest news scraped from multiple sources. The sentiment scores are used by the Pricing Engine to skew prices.

3.5 Market Data Feed

This module provides the Pricing Engine with the latest market data necessary for pricing instruments.

4. Modularity and Future Improvements

The system is designed with modularity in mind, meaning it's easy to replace or add components while maintaining the protocols between them. Although currently in its alpha version, future enhancements could include:

- Improve the pricing engine's functionality to cover more complex scenarios and diverse instruments.
- Enhance the Sentiment Analysis Engine by connecting it to more news sources and refining the sentiment scoring algorithm.
- Improve the GPT-4 LLM's communication abilities to understand a wider array of lexicon and keep track of the context of each conversation.
- Add more market data sources for better pricing.

5. Conclusion

This document provides a high-level design of a Telegram RFQ bot. The bot uses sophisticated machine learning models to automate the RFQ process, delivering value to traders by reducing the time spent answering RFQs.