Trading Strategy Using Embeddings

Ahmed Noor

September 12, 2024

1 Introduction

The trading strategy is designed to predict market movements using pre-calculated vector embeddings derived from historical stock price data. The primary goal is to identify profitable trading signals (buy, hold, sell) based on the embeddings and to backtest the strategy for profit.

2 Mathematical Formulation

2.1 Feature Extraction

Given the vector embeddings for price movements, the following features are extracted:

- Mean of the embedding vectors
- Standard deviation of the embedding vectors

Let $\mathbf{v}_i$ be the embedding vector at time i, then:

$$\text{mean}_i = \frac{1}{d} \sum_{j=1}^{d} v_{i,j}$$

$$\text{std}_i = \sqrt{\frac{1}{d} \sum_{j=1}^{d} (v_{i,j} - \text{mean}_i)^2}$$

where d is the number of dimensions in the embedding vector.

2.2 Feature Importance Extraction

Feature importance is determined using a machine learning model. In this case, XGBoost Regressor is used to identify the top features from the embedding vectors:

$$\text{Feature Importance}_k = \text{Importance of feature } k$$

where features are ranked and selected based on their importance scores.

2.3 Trading Signal Generation

A trading signal is generated based on the mean of the embedding vectors:

$$\text{signal}_i = \begin{cases} 1 & \text{if mean}_i > \text{threshold}_{\text{buy}} \\ -1 & \text{if mean}_i < \text{threshold}_{\text{sell}} \\ 0 & \text{otherwise} \end{cases}$$

where $\text{threshold}_{\text{buy}}$ and $\text{threshold}_{\text{sell}}$ are predefined thresholds.

2.4 Profit Calculation

The profit from the trading strategy is computed based on the trading signals. Let p_i be the price at time i:

$$\text{Profit} = \sum_{i=1}^{N-1} [\text{signal}_i \cdot (p_{i+1} - p_i) - \text{transaction\_cost}]$$

where transaction_cost accounts for the cost of executing trades.

3 Implementation

The strategy involves the following steps:

1. Extract embedding features from the vector embeddings.
2. Generate trading signals based on the extracted features.
3. Backtest the strategy to calculate profit by simulating trades.

3.1 Feature Selection and Model Application

The initial approach involved selecting the top features based on their importance scores obtained from the XGBoost Regressor model. A Random Forest Classifier (RFC) was then trained using these top features. However, it was observed that using only the top features did not yield profitable results.

To improve the strategy, all features including the embedding vectors, along with additional engineered features such as mean, rolling mean, and Bollinger Bands, were used to train the RFC. This comprehensive approach resulted in a more effective model, demonstrating better performance.

3.2 Backtesting

In backtesting, trading decisions are simulated to calculate the total profit. The backtesting includes:

- Entering a trade (buy) when the signal is positive and no open position exists.

- Exiting a trade (sell) when the signal is negative and a position is open.
- Accounting for transaction costs and potential losses (stop-loss) or gains (take-profit).

4 Conclusion

The trading strategy leverages vector embeddings to extract meaningful features that inform trading decisions. The effectiveness of the strategy is evaluated based on simulated trading profits and performance metrics. It was found that incorporating all features along with engineered features provided better results compared to using only the top features. Due to time constraints, further optimization and refinement were not possible, but the current approach demonstrates the potential of using embeddings for trading strategies.