Thai Tran

Market Sentiment Analysis

1. Choice of dataset

Kaggle Financial Sentiment Dataset: https://www.kaggle.com/datasets/sbhatti/financial-sentiment-analysis

I pick this dataset since it is accessible. It's two datasets (FiQA, Financial PhraseBank) combined into one easy-to-use CSV file. It provides financial sentences with sentiment labels.

2. Methodology

a. Data Preprocessing

The dataset has 3 labels: positive, negative, neutral. The features are financial news headlines. We will first clean the text by converting it to lowercase, removing special characters, punctuation, and stopwords. Then, by applying tokenization and lemmatization, we break sentences into words and reduce words to their base form. Lastly, we will convert text into numerical representations using word embeddings (BERT) for model training. **BERT** is a deep learning NLP model that is pre-trained on large datasets.

b. Machine learning model

We want to predict the sentiment (positive, negative, neutral) of financial news headlines using the dataset. I choose **FinBert** as my machine learning model although we haven't seen it in class. It is extremely powerful compared to other models since it has context-aware embeddings and requires minimal fine-tuning. To implement it, at least, how I saw other do it: first, we tokenize news headlines using BERT tokenizer, then we fine-tune using **PyTorch** and **TensorFlow**. This model requires a lot of "computational resources", which is perfect since I am getting a new laptop with 32GB RAM and a RTX 4070 GPU.

c. Evaluation Metric

I will use confusion matrix to demonstrate the model's performance (**accuracy, precision, and recall**) by listing the number of correct and incorrect predictions for each sentiment category. I choose to set a baseline of 70% accuracy.

I am also considering using **F1 score**, which calculates the harmonic mean of Precision and Recall to prevent the model from predicting too many neutral values and ignoring positive and negative ones (the minorities). **Log Loss** is a must as well, since it will penalize overconfident wrong predictions, thus we aim for a log loss for calibrated predictions.

3. Application

The user enters a stock ticker and the desired period for the analysis. Then, the application will automatically fetch related news headlines for sentiment analysis. After submission, the user should receive a sentiment classification (positive, negative, or neutral) with a confidence score for each headline. Additionally, it will display a sentiment trend graph over the chosen period.