Using Intra-Stock, Inter-Stock, and Temporal Aggregation to Predict Stock Market Returns and Stock Selections

Money Is All You Need

Prottoya Chowdhury (pchowdh4), Bailing Hou (bhou14), Devesh Kumar (dkumar23), Haoyang Xu (hxu126)

Purpose

Problem: Stock price forecasting is a challenging problem due to the high volatility of the market

Solution: We use Transformers to train a deep learning model to predict a series of future return data for different stocks

Introduction

Reference Paper: Tong Li, Zhaoyang Liu, Yanyan Shen, Xue Wang, Haokun Chen, and Sen Huang. Master: Market-guided stock transformer for stock price forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38:162–170, Mar. 2024.

As suggested by our team name, *Money Is All You Need*, we are interested in the financial market, and stock price prediction is an important topic within this field with ongoing research. The reference paper uses **deep learning** techniques and innovative methods that capture the stock correlation and automatically select relevant features with **market information**, making it an outstanding application of deep learning to tackle real-world challenges effectively and creatively.

We aim to predict **U.S. stock prices** by training on the **Global Factor Dataset** and solve limitations of existing works through incorporating a **modified Transformer model** (using market-guided gating, intra-stock aggregation, inter-stock aggregation, temporal aggregation, and prediction layers) to tackle this problem elegantly. Since the purpose of this model is to predict a series of future return data for different stocks, the task of the model would be a classified as a structured prediction problem.

Methodology

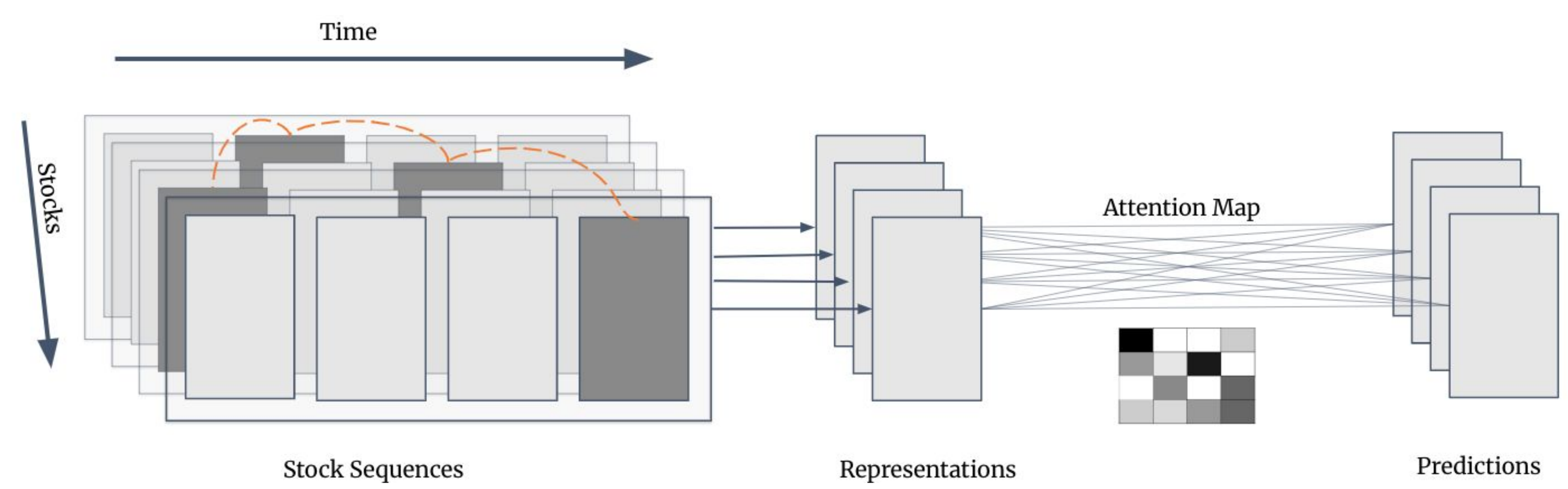

Figure 1: Traditional Stock Model Architecture

Among other flaws, traditional stock models like the one shown above cannot factor in real-time stock interactions. The diagram of our improved model is shown below:

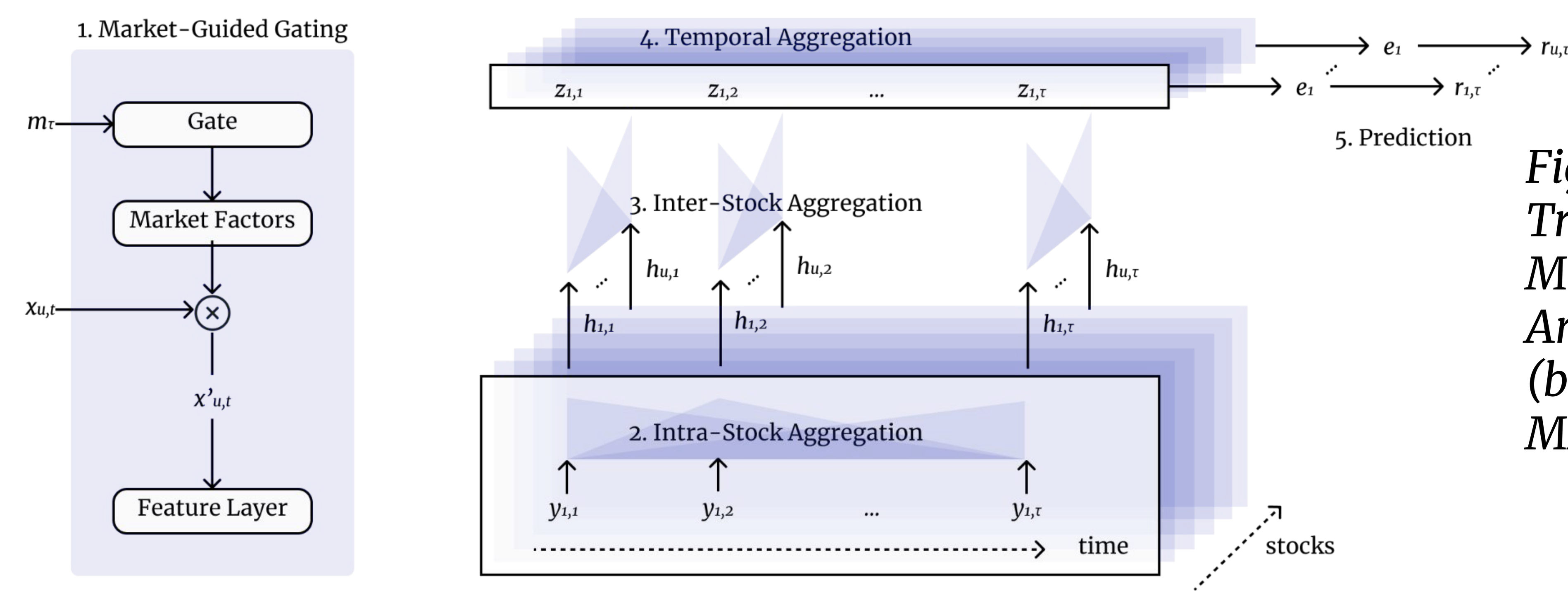

Figure 2: Transformer Model Architecture (based off of MASTER)

There are **5** key components of our Transformer model. The **market-guided gating** element uses a market status vector to scale different features based upon the market status vector that determines what features are the most important. **Intra-stock aggregation** creates an embedding that takes into account the temporal context around a *specific point in time for a stock*. **Inter-stock aggregation** involves using the overall market dynamics to create an embedding of how *stocks compare to each other*. This step uses *attention*. **Temporal aggregation** means putting temporal embeddings together to form an aggregated stock embedding. The final **prediction layer** is a series of dense layers to help determine what the returns for a stock will be.

Results

Information Coefficient (IC)	0.0082
Rank Information Coefficient (RankIC)	0.0953
Information Ratio based IC (ICIR)	0.0271
Information Ratio based RankIC (RankICIR)	0.5296

Table 1: Evaluation of Model with performance benchmarks

Information coefficient is the correlation between the predicted returns and the true returns. RankIC is similar to IC but uses rank, thus prioritizing the direction rather than magnitude. ICIR is a normalized version of IC. RankICIR normalizes RankIC.

Our model achieved an average MSE loss of around **1**, which indicated that the model's prediction of monthly stock returns differed from the actual returns by about **1** standard deviation away. We also chose to investigate our results further by dynamically creating **portfolios** of stocks that consisted of the top third of stocks the model predicted as having the highest returns. We observed returns for these portfolios and found that on average they gained about **0.3%** in a single month while maintaining an annual sharpe ratio of about **0.35**, which is slightly low and suggests a bit of high volatility for the amount of returning earned.

This leads us to conclude that the model is able to identify and learn features that are predictive of stock returns for **some** stocks but not others. For stocks that the model predicts as having positive returns, there seems to be some degree of accuracy.

Discussion

One lesson we learned was that training a model on stock data requires a lot more computations than what we expected. Our solution was to let the model run on **OSCAR** for long enough to get the task done, but, in the future, we hope to find ways to further optimize this process.

We trained our model mostly using data that is specific to a particular company. However, market and economic conditions strongly affect stock returns, and different types of companies might be impacted differently by **market conditions** (for example, during events such as the Great Depression, the COVID-19 Pandemic, or the surge of AI currently). Finding a way to merge our dataset with more relevant global economic, market, and social information would likely enhance our results. A lot of our dataset contains values that are missing, so, for the purpose of training our model, we dropped these pieces of data. Finding a strategy to integrate these pieces of information without biasing our results could introduce more complexity to our model. Those could all be possible ideas to further implement in future works related to this project.