Temporal Relational Ranking for Stock Prediction Summary

Yaxin Li

Feb 25, 2020

Summary of The Artical

This article first analyzed the drawbacks of mainstream solutions by using time series for stock predictive, this article then illustrated the advanced techniques like deep neural networks. However, there are limitations of existing neural network-based solutions, so this article *proposed Relational Stock Ranking* (RSR), which integrates stock relation into prediction. Their work is directly related to stock prediction, graph-based learning, and knowledge graph embedding.

The authors formulated stock prediction as a ranking task and demonstrated the potential of learning-to-rank methods for predicting stocks. The core of the framework is a neural network modeling component, named Temporal Graph Convolution, which can handle the impact between different stocks by encoding stock relations in a time-sensitive way. on NASDAQ and NYSE demonstrate the effectiveness of the solution — with three different back-testing strategies, the RSR framework outperforms the SP 500 Index with a significantly higher return ratio. The experimental of extensive back-testing results demonstrate that the proposed RSR model significantly outperforms SFM (one existing neural network-based model) with more than 115% improvements in return ratio.

After that, the authors explored the potential of emphasizing top-ranked entities with more advanced learning-to-rank techniques. In addition, they integrated risk management techniques in finance into the RSR framework to force the predictions to be risk-sensitive. Furthermore, they investigated the performance of RSR under multiple investment operations such as buy-hold-sell (aka. long position) and borrow-sell-buy (aka. short position). Moreover, the article will further integrate alternative data such as financial news and social media contents into the predictive model.

Summary of The Github Link

There are three python documents in Github. The first one is the knowledge about LSTM and graph-based learning, which forms the building blocks of their method.

The second one is for data collection. They collect the stocks from the NASDAQ and NYSE markets that have transaction records between 01/02/2013 and 12/08/2017, obtaining 3, 274 and 3, 163 stocks respectively.

The third one is about Parameter Settings. They implement the models with TensorFlow except SFM of which use the original implementation.

It employs a grid search to select the optimal hyperparameters regarding IRR for all methods. For SFM, it follows the original setting in, optimizing it by RMSProp with a learning rate of 0.5, and tuning the number of frequencies and hidden units within 5, 10, 15 and 10, 20, 30, respectively.

For all other methods, it applied the Adam optimizer with a learning rate of 0.001. It tune two hyperparameters for LSTM, the length of sequential input S and the number of hidden units U, within 2, 4, 8, 16 and 16, 32, 64, 128, respectively. Besides S and U, it further tuned in Prediction Layer, the equation 8 in the article, which balances the point-wise and pair-wise terms; specifically, it tuned within 0.1, 1, 10 for $Rank_{LSTM}, GCN, RSR_E, and RSR_I$. It further tuned the of the regularization term in GBR within 0.1, 1, 10.