Transformers and Multi-features Time2Vec for Financial Prediction

Bui Nguyen Kim Hai, Nguyen Duy Chien

TDK CONFERENCE – IT SCIENCE SECTION, 2024 SPRING

Budapest, Hungary
May 29, 2024

Outline

1. Introduction
   - Motivation
   - Related work
2. Proposed model and techniques
   - Data collection
   - Preprocessing data
   - Model architecture
   - Decoding engineering
3. Results and Conclusion
4. Summary

Outline

1. Introduction
   - Motivation
   - Related work
2. Proposed model and techniques
   - Data collection
   - Preprocessing data
   - Model architecture
   - Decoding engineering
3. Results and Conclusion
4. Summary

Motivation

By other works

- Researchers try to combine Time2Vec with CNN, RNN, LSTM, and Attention mechanism
- For instances:
  - Aeroengine Risk Assessment
  - Predicting Production in Shale and Sandstone Gas Reservoirs
  - Stock Price Forecasting

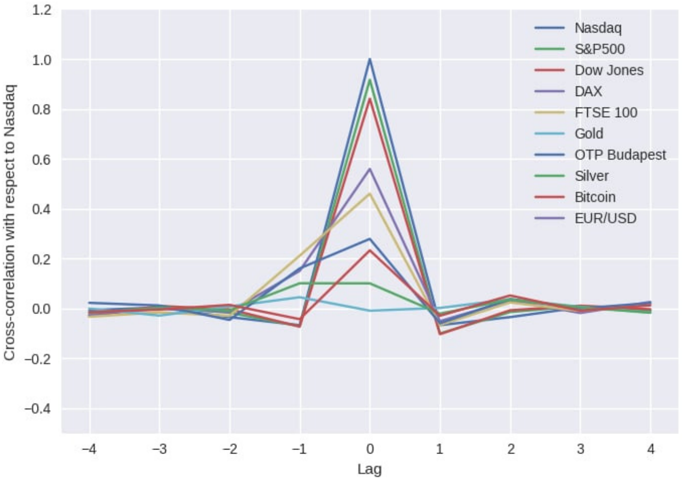
In finance area

- Studies primarily rely on one dataset

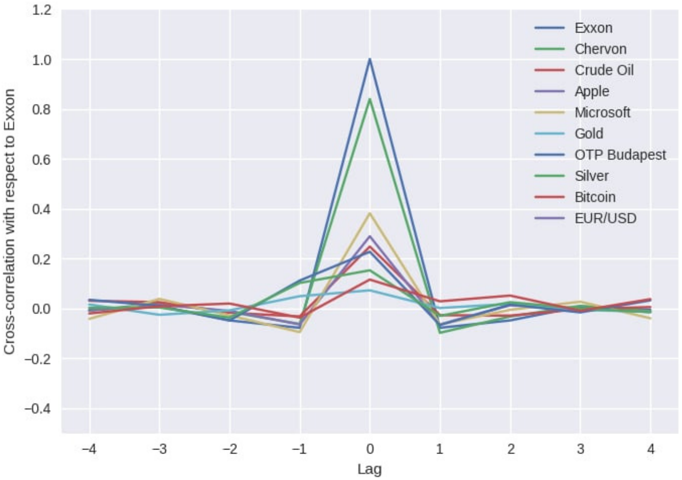
By observing trends

- Stock's trend is a Markov process
- Historical data offers limited foresight
- Stocks having similar trend is more promising

Motivation: Cross-correlation to NASDAQ



Motivation: Cross-correlation to Exxon Mobil



Related work

ARIMA

Making one-step-ahead predictions

RNN

Handling temporal problems in sequential data and time-series analysis.

LSTM

Using gates, LSTM enables network to learn long-term dependencies and prevent the vanishing gradient problem.

Transformer

The SOTA architecture that works well in many area such as NLP, and time-series

Time2Vec

Use to embed the time-series data to vector

Outline

1 Introduction
- Motivation
- Related work

2 Proposed model and techniques
- Data collection
- Preprocessing data
- Model architecture
- Decoding engineering

3 Results and Conclusion

4 Summary

Data collection

Where to collect?

Yahoo Finance

What will be collected?

Date, Open, High, Low, Close, Volume columns

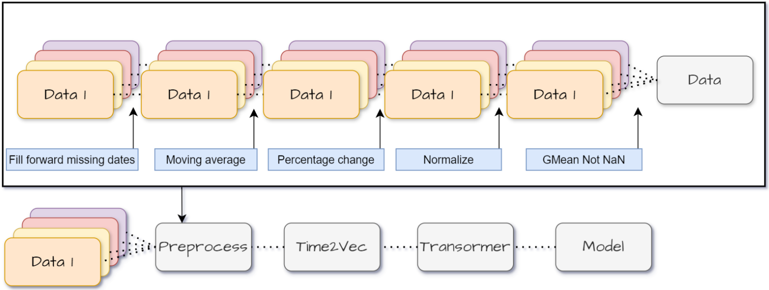
How many datasets should we collect?

Two, three, four ..., as long as they are highly correlated to each other

Collected datasets

- Group1: NASDAQ, S&P500, DJI, DAX
- Group2: Exxon Mobil, Chervon

Preprocessing data: The pipeline

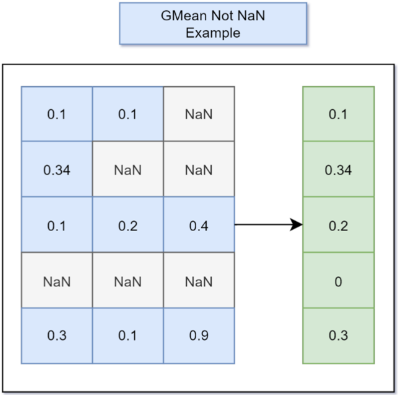


The preprocessing data pipeline.

Techniques

- Fill-forward: Filling missing data in dataset
- Moving Average: Smoothing dataset by averaging data
- Percentage Change: Compute the difference in the data
- Min-Max Normalization: Normalizing dataset
- Geometry Mean Not NaN (GMNN): Combining multiple datasets

Preprocessing data: But... What is GMNN?

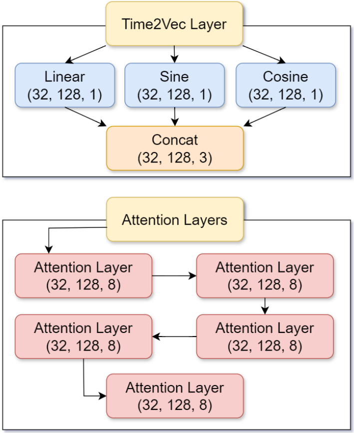


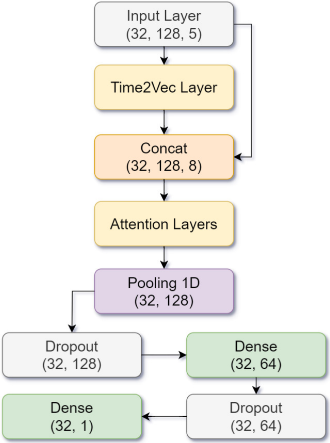
GMNN Attributes

- Union: Handling length difference when combining datasets
- Invariant: Keeping the data stays normalized
- Representation: The output reflects the whole datasets

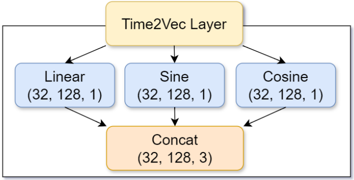
A simple sample of applying GMNN transformation

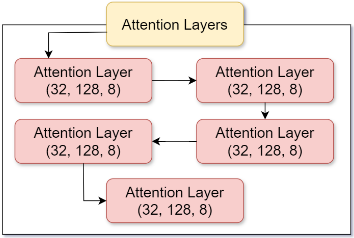
Model architecture: Proposed model





Model architecture: Role of layers





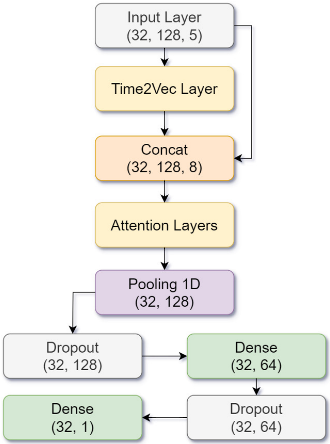
Roles

- Time2Vec
  - Linear: Capturing linear trends
  - Sine, Cosine: Encoding positions and capturing periodic behaviors
  - Concat: Concatenating above three layers
- Attention Layers
  - To study the trend from different aspects, positions

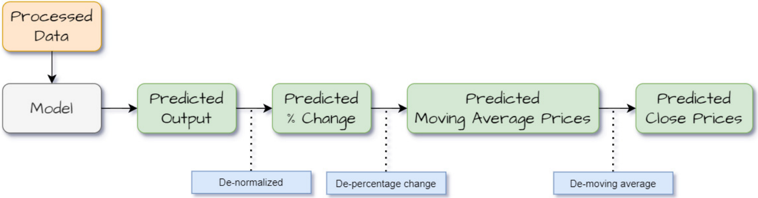
Model architecture: Role of layers

Roles

- Time2Vec: Catch continuous attribute of time
- Concat: Apply Residual Connection
- Attention: Deep understanding trend movements
- Pooling: Reducing dimension
- Dropout: Prevent over-fitting
- Dense: Apply activation functions (ReLU)



Decoding engineering



The decoding pipeline.

Techniques

- De-normalized
- De-percentage change
- De-moving average

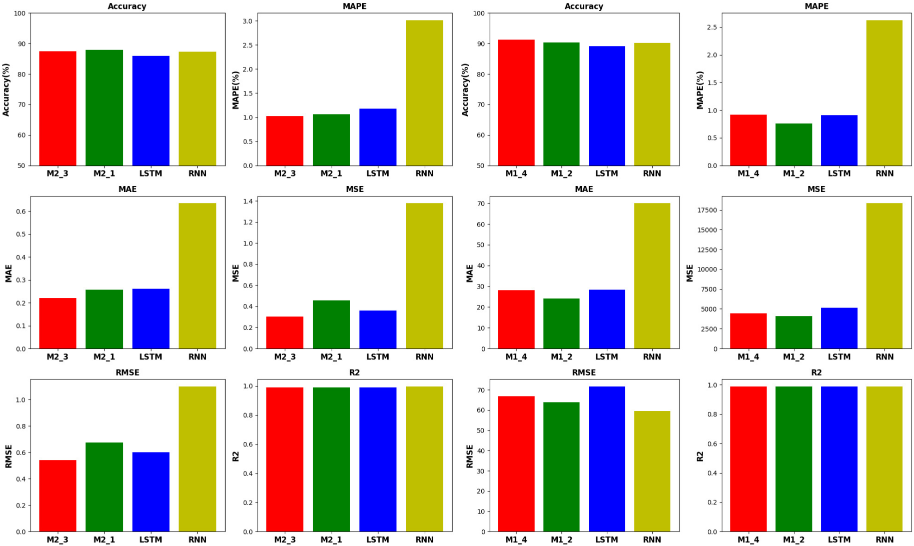
Why don't we use De-GMNN step?

- Output is **normalized** (Invariant)
- Target is **one** dataset, output only reflects that one

Outline

1. Introduction
   - Motivation
   - Related work
2. Proposed model and techniques
   - Data collection
   - Preprocessing data
   - Model architecture
   - Decoding engineering
3. Results and Conclusion
4. Summary

Results



Comparing 6 metrics with respect to Exxon (Left), NASDAQ (Right)

Conclusion

Conclusion

By leveraging multiple criteria to evaluate the proposed model such as

- MAE, MAPE, RMSE, MSE, R2-score (price prediction task)
- Accuracy (trend forecasting task)

We can proudly say that, the multi-feature model

- **Outperforms** the single-feature one in most cases and they are **extremely close** to each other in other scenarios.
- Usually yields **better** result than the SOTA in almost every contexts.

Outline

1. Introduction
   - Motivation
   - Related work
2. Proposed model and techniques
   - Data collection
   - Preprocessing data
   - Model architecture
   - Decoding engineering
3. Results and Conclusion
4. Summary

Summary

Summary

- We explore deep learning for challenging stock price prediction
- Paving the way for new feature studies and applications in various deep learning models
- Demonstrates correlation-based features and innovative neural networks improve stock price prediction

Further Research

- Fine-tuning the architecture
- Continuing improving processing methods
- Comparing to other SOTA neural networks like KAN
- Applying the architecture to other areas

Thank you for your attention!