

ZLF's Solution in the MITSUI&CO. Commodity Prediction Challenge

A. MODEL SUMMARY

A1. Background on me/my team

- Competition Name: MITSUI&CO. Commodity Prediction Challenge
- Team Name: ZLF
- Private Leaderboard Score: 0.532
- Private Leaderboard Place: 5th

Team member

- Name: Zhu Linfeng
- Email: zhulinfeng04@outlook.com

A2. Background on me/my team

- What is your academic/professional background?
I'm a 4th year student at the University of Electronic Science and Technology of China and the University of Glasgow.
- Did you have any prior experience that helped you succeed in this competition?
No.
- What made you decide to enter this competition?
I'm interested in finance/investment and time series modeling.
- How much time did you spend on the competition?
Maybe 2-3 weeks in total.

A3. Summary

I used two models for prediction: a 4-day window RNN model and a single-day MLP model. It was observed that the RNN model produced relatively stable and conservative outputs, whereas the MLP model exhibited greater volatility in its predictions. The final prediction is the average of the outputs from both models. In addition, raw features were directly used without additional processing, with missing values uniformly filled with -1. During training, I employed a composite loss function combining MSE and ranking loss (from @takeshimorimura's notebook), implemented in PyTorch on Kaggle's T4 GPU environment. Each model took about 3-4 minutes to train.

A4. Features Selection / Engineering

No specific feature selection/engineering was performed, all raw variables were used. Missing values were filled with -1, and data normalization was handled by the built-in

LayerNorm layers in the models.

A5. Training Method(s)

I used two independent models for training and prediction: a 4-day time window RNN model and a single-day MLP model. A short window was chosen because experiments showed that longer windows did not lead to performance improvements and could even introduce noise.

Both models were trained end-to-end using the same composite loss function, which was adapted from takeshimorimura's baseline and combined with a learning rate decay strategy.

The final prediction result is the simple average of the outputs from the two models, aiming to leverage their distinct predictive characteristics.

A6. Interesting findings

- What was the most important trick you used?

The trick was building a dual-model ensemble system and performing mean ensembling of their outputs, combining the characteristics of both models. Additionally, a hybrid loss function was used to balance both the absolute accuracy and relative ranking of the predictions.

- What do you think set you apart from others in the competition?

The key differentiator likely lies in the complementary integration of two models with distinctly different predictive behaviors. Training observations revealed that the RNN predictions were relatively stable and conservative, while the MLP predictions were more volatile and sometimes more accurate. This simple averaging of "conservative" and "aggressive" approaches yielded more robust results than using a single model alone.

- Did you find any interesting relationships in the data that don't fit in the sections above?

It was found that in this task, short-term historical information appears to be more valuable for prediction than long-term information. In experiments with different time windows, longer windows did not improve performance and even introduced noise. Therefore, a short window of only 4 days was ultimately selected, and a simple RNN was adopted.

However, this speculation is primarily based on limited experimental observations, and its validity still requires further rigorous verification.

A7. Simple Features and Methods

- What would the simplified model score?

The private score of the single RNN model is 0.509

A8. Model Execution Time

- How long does it take to train your model?

3-4 min

- How long does it take to generate predictions using your model?
in seconds
- How long does it take to train the simplified model (referenced in section A6)?
3-4 min
- How long does it take to generate predictions from the simplified model?
in seconds

A9. References

- @takeshimorimura 's notebook: <https://www.kaggle.com/code/takeshimorimura/mlp-baseline-cv>