

# Quantitative trading based on AlphaNet

## Abstract

Inspired by the CNN and RNN, we propose a new quantitative trading model. This model overcomes the problem of poor performance of traditional convolutional network models on time series, especially financial data series, by combining GRU and transformer with CNN model. We get good prediction results on the SP100 dataset after 2015.

## Motivation

The key to the success of our model is to combine the convolutional network and the sequence model. Since CNN is used to extract features in image classification, and RNN can learn the relationship in time, we think that we can transform financial data into "data picture", input them into CNN to extract features and use sequence model makes predictions.

## Methods

- **CNN** Use the data image as the input to the CNN. The data for each stock forms a data image from  $t - n$  to  $t$ . If we have  $n$  stocks in an interval, we can get  $n$  images of the data for that interval. In the data image, each column represents a time step (such as a day), and different rows represent different types of values. Such as opening price, highest price, lowest price, closing price, one-day yield, etc.

$open(t - n)$	...	$open(t - 1)$	$open(t)$
$high(t - n)$	...	$high(t - 1)$	$high(t)$
$low(t - n)$	...	$low(t - 1)$	$low(t)$
$close(t - n)$	...	$close(t - 1)$	$close(t)$
$volume(t - n)$	...	$volume(t - 1)$	$volume(t)$
Some other features			

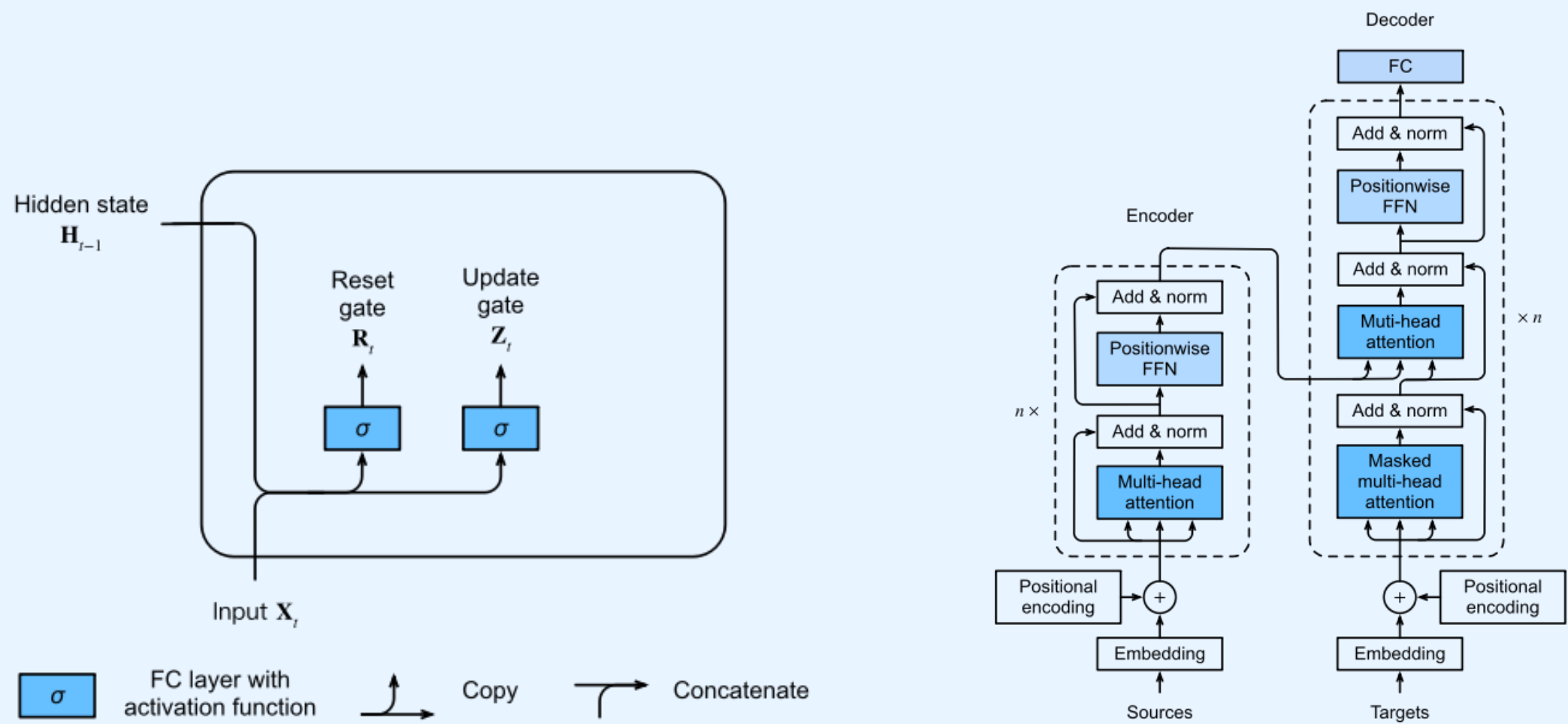


Figure 1. GRU and Transformer

- **GRU** a gating mechanism in recurrent neural networks, the features extracted by CNN are input into the GRU layer. In practical attempts, GRU can be replaced with other sequence models such as RNN, LSTM and its variants to improve performance.
- **Transformer** Compared with the GRU model, the Transformer uses a self-attention mechanism to differentially weight the importance of each part of the input data, and performs best in our experiments

## Models

After preprocessing the data image, the input consists of a feature extraction layer, which consists of multiple nonlinear functions and pooling layers. The nonlinear function functions include correlation coefficient, maximum value, minimum value, etc. The convolution kernels in CNN are consistent. Two feature extraction layers are used in our model, and the functions of the two layers have different lookback intervals (10 days and 5 days).

The extracted features are input to the pooling layer and the output is flattened and input to the recurrent neural network layer. Finally, the fully connected layer is input to obtain the prediction result. The parameters of the entire network will be optimized by back-propagation according to the loss of the prediction target. The role of the fully connected layer is to perform weighted synthesis of features, which is consistent with the fully connected layer in CNN.

Name	Definition
$corr(X, Y, d)$	Correlation coefficient between the series $X$ and $Y$ for the past $d$ days
$return(X, d)$	$(X - delay(X, d)) / delay(X, d) - 1$ , $delay(X, d)$ is $X$ over the past $d$ days
$zscore(X, d)$	The mean divided by the standard deviation of $X$ over the past $d$ days
$min(X, d)$	The minimum value of $X$ over the past $d$ days
...	...

Table 1. Functions in Feature Extraction Layer

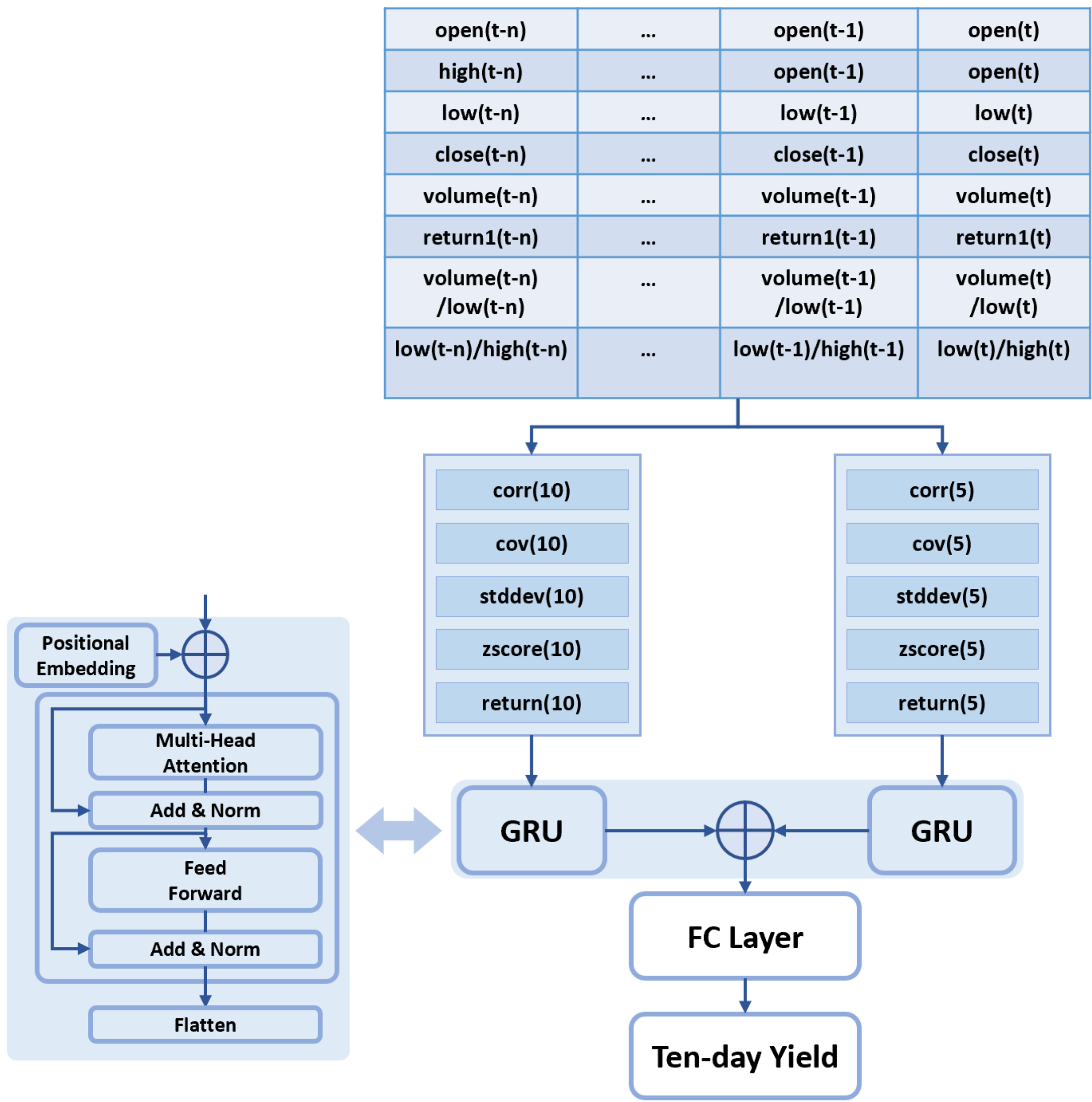


Figure 2. The structure of model

## Results

We have trained on multiple stocks. The selected training set starts from July 2015, and the test set starts from January 2021. The following figure shows the prediction results of the stocks Tesla and Facebook. It can be seen that using transformers The model is better at predicting both trends and concrete values than the model using GRU. We tested an average RMSE of 0.0329 for the transformer module and 0.1138 for the GRU module.

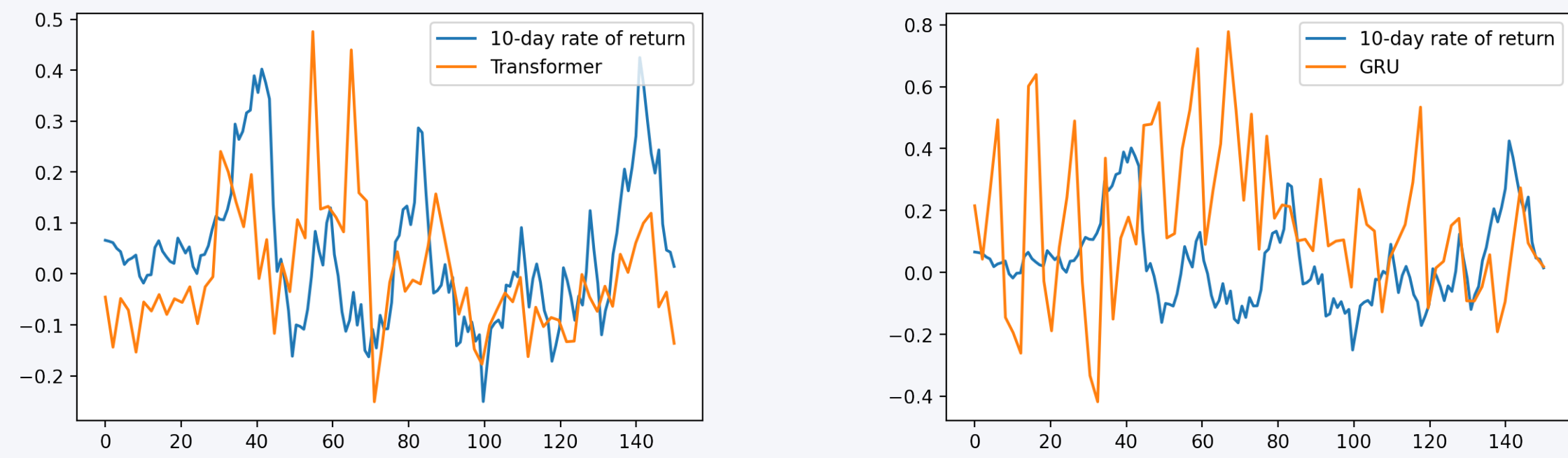


Figure 3. Test results using Transformer and GRU, starting on 2021/01/08

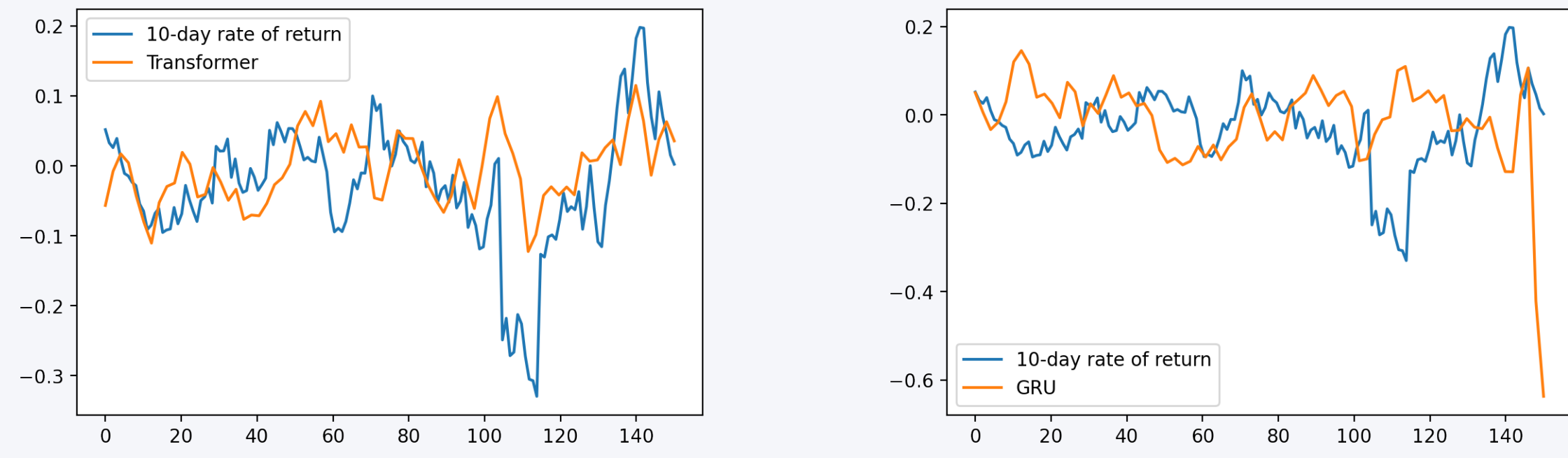


Figure 4. Test results using Transformer and GRU, starting on 2021/07/12

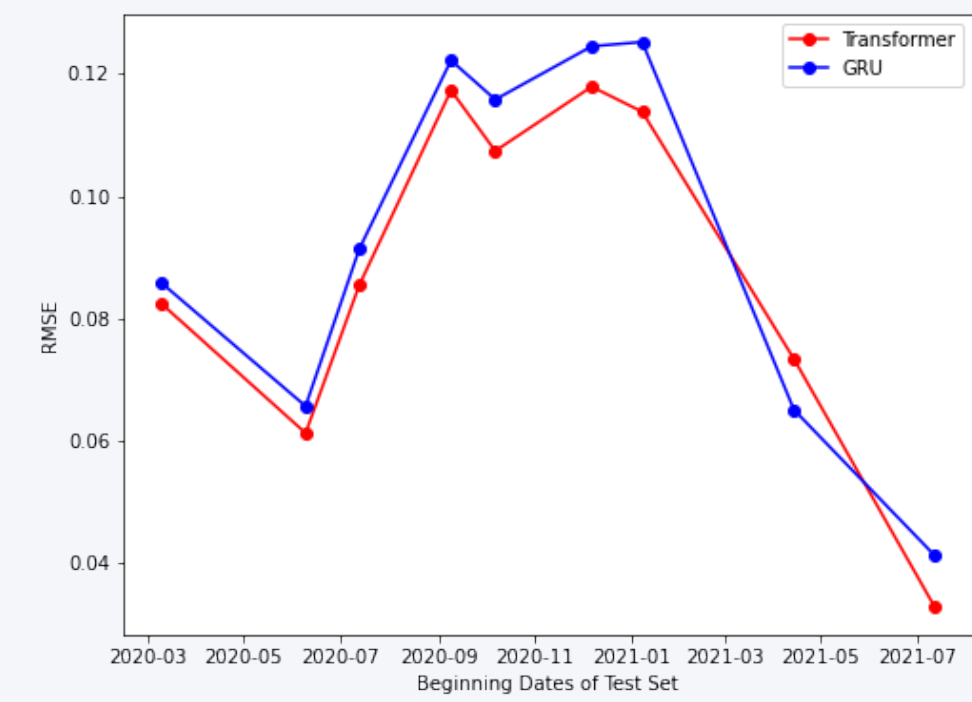


Figure 5. RMSE

Test set	Model	RMSE
2021/1	Transformer	0.1138
2021/1	GRU	0.1252
2021/7	Transformer	0.0329
2021/7	GRU	0.0413

## Evaluation

Compared with the common genetic programming + random forest model in the industry.

### Advantages

1. End-to-end learning, optimized using the same objective function
2. Training process is simplified without manual intervention, avoiding the work of selecting features

### Disadvantages

1. The model interpretability is low.
2. The feature extraction layer that can be embedded in the neural network is still relatively limited, and the factor calculation method cannot be updated in practical applications.