EasyChair Preprint

№ 10141

Trend Forecasting in Financial Time Series Using a Combinational Method of Heuristic Pattern Recognition and Support Vector Machine

Fatemeh Khazaeni and Mohammad Amin Shayegan

May 12, 2023

Trend forecasting in financial time series using a combinational method of heuristic pattern recognition and support vector machine

Fatemeh Khazaeni[1] , Mohammad Amin Shayegan [2,*]

[1] Department of Computer, Shiraz Branch, Islamic Azad University, Shiraz, Iran
Email: Fatima.Khazaeni@gmail.com

[2] Department of Computer, Shiraz Branch, Islamic Azad University, Shiraz, Iran
Email: MA.Shayegan@iau.ac.ir
*: Corresponding Author

Abstract. Whereas many studies have been done on forecasting different time series, it has always been associated with challenges such as uncertainty. For example, in financial time series, if we want to predict the price, due to the Non-stationary nature of the time series, forecasting will face false regression. To solve this problem, in this research, instead of price forecasting, trend forecasting has been done. In this case, since the subtraction operator has been used to calculate the trend, the effect of Non Stationary nature is removed and the issue of false regression is solved. In this research, using machine learning methods through the return forecasting approach, the trend in financial time series has been predicted.

In this research, the effective features of the data of the last 10 years in the foreign stock market for the shares of several different companies have been examined and compared with the Benchmark index of the market, and after creating different machine learning models and maximizing the accuracy of the results, a satisfying application has been extracted to be used as an effective trading tool for traders. To train the model, random forests algorithm and Support Vector Machine, Feature Selection and Heuristic algorithms have been used.

The evaluation of this model on the Foreign Stock market in recent years shows that not only the presented system can make effective predictions, but it is largely robust to market fluctuations and performs better than other existing methods.

Keywords: Forecasting, Financial time series, Machine learning, Trend Prediction

Introduction

So far, many studies have been done on the prediction of time series data, for example [1] used SVM to predict the daily path of stock price movements in Korea with a success rate of 56%. [2] used sequential pattern mining techniques to predict the share price and reached a success rate of 56%. [3] combined decision tree and neural network to predict the Taiwan stock market. Although their test data set was relatively small and only included several shares, the accuracy of their combined model reaches about 70%. According to some experimental studies [4], the prediction accuracy of several machine learning models (including C4.5, K*, logistic model tree, etc.) is in the range of 48% to 54%. In [5], LSTM neural network was used and the accuracy reached 55%, and [6] used a convolutional neural network for prediction, and despite increasing the accuracy to 77%, it still does not perform well in some cases. All the mentioned studies have a common weakness and their availability in practice is still questioned. In their studies, a small amount of carefully selected and labeled stock data was used to train and test the model. Since the data does not cover all stocks and their movements in a stock market, the generalization of the model in real applications is reduced.

The proposed system presented in this research is a modular system that includes different parts. The basic problem of this research is the prediction of stock market price movements.

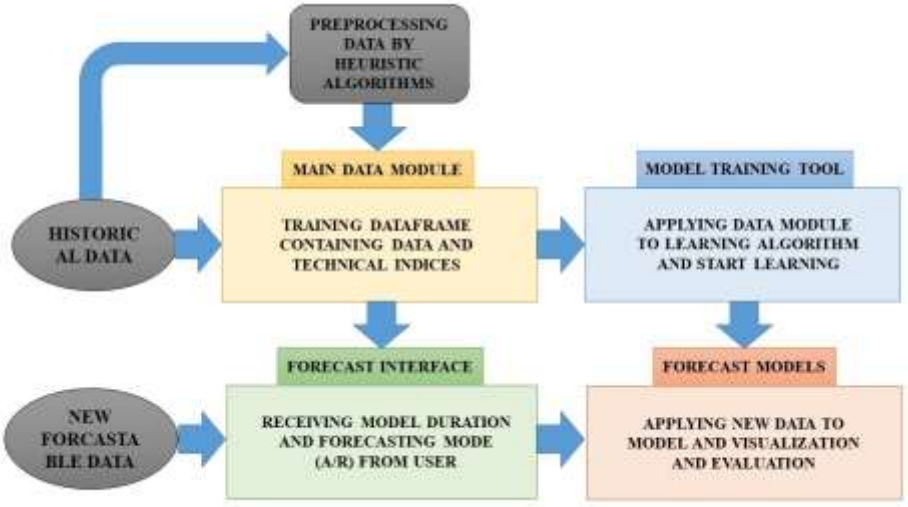

Fig.1. Representation of the system's architecture

The main contribution of this research is the use of heuristic pattern recognition techniques and structural inspection to simulate the human perspective of technical analysts, which along with the SVM learning algorithm, balancing techniques, and the feature selection method has increased the accuracy of the algorithm performance.

As shown in fig 1, the system designed and implemented in this research includes the following:

• Preprocessing of historical data behavior with a heuristic algorithm

• Forming the main data module including data and technical indicators

• Model training tool and data module application to the learning algorithm and training

• Interface to receive basic information from the user to predict

• Forecasting and visualization of results and system evaluation

Methodology

In time series analysis, the main goal is to create a statistical model for time-dependent data, based on its historical data and this makes it possible to predict the future of the discussed phenomenon. In other words, time series analysis is creating a retrospective model to enable future decisions. The creation and application of statistical and stochastic models in the form of time series analysis have become very widespread today with the help of intelligent algorithms, and the result is models that, with flexible parameters, can predict the future using past parameters. This phenomenon also works in financial time series, including Commodity Stock Market, Foreign Exchange Market, and Cryptocurrencies.

In this research, we are specifically focused on trend changes. Because it determines the general direction of the market in the asset price chart and is considered a powerful auxiliary tool for traders.

Defining the meaning of the term "Trend" in [7] It has been stated that the long-term trend or tendency is the evolution of the studied variable in a long-term period, without taking into account the periodic, seasonal, and irregular changes. In this case, we ignore the time series fluctuations and pay attention to their overview.

Because we intend to predict the price movements of a share at the end of the predefined period, we define four main classes based on the shapes of the closing prices of a share.

Price behavior falls into one of the following categories:

1. Rises (Increasing class)

2. Drops down (Decreasing class)

3. Remains almost constant (Unchanged class)

4. Oscillates with high amplitudes (Unknown class)

Please note that as the most important index, we consider Closing prices of the daily price chart as the main comparative criterion.

In addition to changes in closing prices, we are also interested in calculating the relative return, which is the return achieved by an asset for a period of time toward the Benchmark.

For a period of n days, the logarithmic relative return can be calculated as follows:

$$rt = \sum_{i=1}^{n}(\ln(1 + f_i) - (\ln(1 + b_i)) \quad (10)$$

where f_i and b_i are the return of the asset and the return of the benchmark on the i-th day, respectively.

In this research, we built the prediction model based on two different modes, the relative mode, and the absolute mode.

In relative mode, relative returns are used instead of closing prices to identify stock patterns and assign labels. Because investors are often interested in outperforming the average(Benchmark).

To predict the trend of time series, historical data pre-processing is done by the use of heuristic algorithms to involve morphological patterns in learning features to simulate the trades of technical analysts. This is processed after segmenting the data with the sliding window method and matching the pattern of the initial part of each window with the predefined patterns to recognize the window's total trend. Then using asset price data including opening and closing prices, high and low prices, the volume of transactions, and adjusted closing prices, as well as calculated technical indicators including simple moving average, exponential moving average, relative strength index, rate of change, and stochastic indicator to form the training matrix and then determine the targets with the calculation of relative return formula. Now the main data module is ready and we apply it to the learning algorithm. Here we have used the data balancing algorithm and then the SVM algorithm to train the data.

The data balancing algorithm module is necessary because there are many windows labeled as unknown due to the majority of large fluctuations of the price movements that have caused unbalanced data.

Usually, for each trading day in the training sample, we use price details, the closing price, the opening price, the high price, and the low price to train the learning models. If the interval of the sample model is 6, its feature dimension becomes 24. Since the features of the training sample can be developed by adding the technical indicators, this might cause high dimensionality and causes some delays. It's not apparent that other information such as volume and technical indicators have a positive effect on the learning performance of classifiers, feature selection techniques can be used to adjust their performance. This method has been widely used in stock forecasting systems. [8]

Although many feature selection methods can be used, finding an optimal combination of features is still an NP-hard problem. In our system, we used the Forward Sequential Search method, which selects one of all candidates for the current state. This method is repeated and the first candidate that was selected is no longer possible to go back. This method does not guarantee an optimal result, but it has a high search speed. If the total length of the sequence is n, the number of search steps is bounded to O_n. Because our feature selection operates on one instance per working day, we need to modify this method slightly, which is that each time a candidate is selected, the features of the model interval will be added.

The characteristics of each trading day in a training sample start with closing, opening, and high and low prices. The next step is as follows:

$$F_i' \coloneqq \{F_i \cup f_i^{(k)} \in \mathcal{F}_i \backslash F_i | J\left(F_i \cup f_i^{(k)}\right) is\ bigger\} \tag{10}$$

where $\mathcal{F}_i$ is the complete set of characteristics of the trading day *i* and *J* is an evaluation criterion. Here, we define the experimental risk of the trained model as *J* criterion and with the following formula:

$$J \equiv R_{emp}(H) = \sum_{i=1}^{N} \mathbb{I}(H(\tilde{\upsilon}_i), y_i) \tag{10}$$

where $\mathbb{I}()$ is an index function.

In this research, we have used the SVM algorithm to train the data. The advantage of using the SVM is that it does not get stuck in local maximum while being simple and fast, and it also works well for high-dimensional data [11].

This method is one of the moderately new methods that outperformed other classification methods in recent years. To classify data with high complexity, we use the *phi* function to move the data to a space with much higher dimensions. To be able to solve the problem of very high dimensions using these methods, from Lagrange's dual theorem to convert the desired minimization problem to its dual form, where instead of the complex function *phi* that takes us to a high-dimensional space, we use a simpler function called the kernel function, which is the vector multiplication of the *phi* function. [12] Different kernel functions can be used, including Exponential, Polynomial, and Gaussian kernels. In this research, we used two types of Polynomial and Gaussian kernel functions. The prediction results obtained for SVM with the Gaussian function are more accurate.

Evaluation

To evaluate the system, we followed the following steps:

• Test settings

• Evaluation of Absolute mode and Relative mode

• Accuracy assessment in SVM algorithm with Polynomial kernel

• Accuracy assessment in SVM algorithm with Gaussian kernel

• Results by applying the Feature Selection algorithm

Considering that our prediction system is based on several fixed prediction intervals, we need to build the learning model by considering feature selection. So we created a dataset for evaluation. For the data set, we randomly take 30% of the samples according to each class for testing and use the remaining 70% for training. The models are trained using the SVM method. Because we set the class distribution to a relatively balanced state, Accuracy is a good measurement criterion for evaluation. In addition, since we randomly split the data 30/70, we repeated the experiment 10 times, and their mean accuracy values were reported. Table 2 shows the results of different algorithms and their accuracies.

Table 1. Comparison of prediction accuracy of different algorithms in two Absolute and Relative modes

Algorithm	Absolute Mode(%)	Relative Mode(%)
SVM (POLY)+FS[1]	**72.19**	**79.87**
SVM (POLY)[2]	**71.26**	**78.81**
SVM (RBF)+FS[3]	**72.19**	**79.87**
SVM (RBF)[4]	**71.52**	**79.87**
R.F.+FS[5]	**67.94**	**76.55**
R.F[6]	**76.29**	**67.81**

As evident in Table 2, the accuracy of the absolute mode is significantly less than the relative mode. In relative mode, the measurement is based on the market index.

In this research, we evaluated the system's performance using the SVM algorithm with two different types of Polynomial kernel and Gaussian kernel. As can be seen, they have very minor differences in practice. In this study, the use of the Gaussian kernel has a better performance than the Polynomial kernel. The Gaussian kernel considers a better category for classes by mapping the data into a radial space.

In general, we have got better results using the feature selection algorithm, although the effect is insignificant for some algorithms, in any case, it has a positive impact on reducing some delays of the algorithm. In the SVM algorithm, this positive effect is significantly observed, and in almost all modes, the accuracy of the trend prediction has improved by up to 1 %. It can be generally concluded that the accuracy of the results algorithm is increased in this method.

The random forests model which has had the best function in recent studies has significantly less accuracy compared to the results of this research.

[1] Support Vector Machine- Polynomial kernel with feature selection

[2] Support Vector Machine- Polynomial kernel without feature selection

[3] Support Vector Machine-RBF kernel with feature selection

[4] Support Vector Machine-RBF kernel without feature selection

[5] Random Forests with feature selection

[6] Random Forests without feature selection

Conclusion

In this study, to predict the trends of time series, we first started with historical data preprocessing. To simulate technical analysts' methods, we used heuristic pattern recognition algorithms. Then, using historical data details including opening and closing prices, high and low prices, traded volume and adjustment price, as well as technical indicators including simple moving average, removable average, relative strength index, the rate of change, and the stochastic indicator we set up a training matrix, and then with the relative return formula, the Target, and the main data module, was prepared and applied to the learning algorithm. In this study, we have used the SVM and data balancing algorithm along with the feature selection method for data training. By completing the training process and determining the parameters of the system, the new data is received and applied to the process of predicting. The performance of our learning models has surpassed many of the existing methods with the criterion of accuracy, due to the use of the combination of the SVM learning algorithm and data balancing and feature selection in an integrated process. As a result, our learning models are also robust to market fluctuations.

Future Work

We can try different methods to improve the system's performance. For example, system parameters can be improved with optimization methods, as well as combining them with other learning algorithms might be effective. In addition, our heuristic algorithms used for pattern recognition can be trained for more patterns.

References

1. Kim, Kyoung-jae. "Financial time series forecasting using support vector machines." *Neurocomputing* 55, no. 1-2 (2003): 307-319.
2. Liang, Mengxia, Shaocong Wu, Xiaolong Wang, and Qingcai Chen. "A stock time series forecasting approach incorporating candlestick patterns and sequence similarity." *Expert Systems with Applications* 205 (2022): 117595.
3. Tsai, Chih F., and Sammy P. Wang. "Stock price forecasting by hybrid machine learning techniques." In *Proceedings of the international multiconference of engineers and computer scientists*, vol. 1, no. 755, p. 60. 2009.
4. Gerlein, Eduardo A., Martin McGinnity, Ammar Belatreche, and Sonya Coleman. "Evaluating machine learning classification for financial trading: An empirical approach." *Expert Systems with Applications* 54 (2016): 193-207.
5. Roostaee, Mohammad Reza, and Ahmad Ali Abin. "Forecasting financial signal for automated trading: An interpretable approach." *Expert Systems with Applications* 211 (2023): 118570.
6. Fang, Zhen, Xu Ma, Huifeng Pan, Guangbing Yang, and Gonzalo R. Arce. "Movement forecasting of financial time series based on adaptive LSTM-BN network." *Expert Systems with Applications* 213 (2023): 119207.
7. Wu, Junran, Ke Xu, Xueyuan Chen, Shangzhe Li, and Jichang Zhao. "Price graphs: Utilizing the structural information of financial time series for stock prediction." *Information Sciences* 588 (2022): 405-424.
8. Zhou, Hongfang, Xiqian Wang, and Rourou Zhu. "Feature selection based on mutual information with correlation coefficient." *Applied Intelligence* (2022): 1-18.
9. Tharwat, Alaa. "Parameter investigation of support vector machine classifier with kernel functions." *Knowledge and Information Systems* 61 (2019): 1269-1302.
10. Zhang, Jing, Shicheng Cui, Yan Xu, Qianmu Li, and Tao Li. "A novel data-driven stock price trend prediction system." *Expert Systems with Applications* 97 (2018): 60-69.
11. Kurani, Akshit, Pavan Doshi, Aarya Vakharia, and Manan Shah. "A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting." *Annals of Data Science* 10, no. 1 (2023): 183-208.
12. Wu, YuRen, Xiang-Jun Shen, Stanley Ebhohimhen Abhadiomhen, Yang Yang, and Ji-Nan Gu. "Kernel ensemble support vector machine with integrated loss in shared parameters space." *Multimedia Tools and Applications* (2022): 1-20.