

Short-Term Stock Price Prediction Algorithm Construction Based on Integrated Learning of SVR and RF with Bagging

Yiqing Liu^{1,#}, Ayiken·Ayitelieke^{2,#}, Jiayi Yu^{3,#,*}

¹ School of Science, Shanghai University, 200444, Shanghai, China

² School of Government Management, Central University of Finance and Economics, 100081, Beijing, China

³ School of Delaware, Southwestern University of Finance and Economics, 610000, Chengdu, China

* Corresponding Author Email: yjy18737797369@163.com

Abstract. In the context of the emergence of artificial intelligence machine learning algorithms, how to handle training data for reasonable and accurate prediction of the stock future market is expected to bring effective methods, but previous traditional models are difficult to be used effectively in complex stock markets. In this paper, by exploring new stock forecasting methods such as Support Vector Regression (SVR), Random Forest (RF), and integration based on integrated learning, the model results are compared with previous traditional model results and the models are evaluated using R2 and MSE metrics. The algorithm based on Bagging integration has better robustness and generalization, in which both R2 and MSE have some improvement compared with those before integration. The research in this paper is beneficial to provide a reasonable prediction approach for stock forecasting later, which can help consumers make better quantitative trading.

Keywords: SVR model, RF model, Bagging integrated learning, SSE index.

1. Introduction

Analyzing the activity patterns of stock market data and predicting its development and changes has always been a hot issue in the financial field. With the continuous development of artificial intelligence and the continuous emergence of machine learning algorithms, configuring the algorithm and processing the training data is an important task. Reasonable and accurate predictions about the future market of stocks are expected to bring high investment returns or hedge risks. The traditional statistical and econometric models are mainly linear regression models, which are difficult to capture the nonlinearity of data in the complex stock market, and have problems such as strong subjectivity, simple models, and unsatisfactory results. Researchers are constantly exploring new stock forecasting methods. With the popularity of artificial intelligence, machine learning algorithms with extremely powerful data processing capabilities and adaptive capabilities, such as artificial neural networks, SVR, RF, and other nonlinear methods have gradually become mainstream.

Yu and Yan [1] designed and optimized a DNN-based model in 2019, and compared the traditional ARIMA linear model, traditional SVR, deep MLP, and LSTM, and the results show that it has higher prediction accuracy. The performance of a single ARIMA and LSTM is not better than that of DNN. In 2020, Bukhari et al. [2] proposed an AFRIMA-LSTM hybrid model, which improved the accuracy of RMSE by about 80%, and outperformed the performance of independent models such as ARIMA, ARFIMA, and GRNN with a MAPE of 0.002%. The popularity of neural networks is gradually increasing. In the same year, Ananthi and Vijayakumar [3] improved the prediction accuracy of securities trading to 85% based on the candlestick chart and K-NN regression algorithm. In addition to neural networks, Nabipour et al. [4] also studied tree-based models and combined neural networks to predict the value of four stock markets. The study found that LSTM has the smallest error and better fitting ability, but there is a bug that runs too long. Also about the tree model, Sadorsky [5] predicted the stock price trend of clean energy ETF last year and found that the prediction accuracy of decision tree Bagging and RF is 85% to 90%, which is about 30% higher than the Logit model. Simian et al. [6] combined the tree-based model with the traditional algorithm, using the reproductive

genetic algorithm to simultaneously optimize the multi-core and SVR models, and the MAPE reached 0.007%.

Previous studies generally considered a single or a few stocks and included their macro, financial, and public opinion factors into the research scope for empirical analysis [7]. However, stock prices are affected by multiple political, economic, and human manipulations. For investors to judge the short-term fluctuations of stock prices, a large amount of this information needs to be collected. In today's information age, stock-related data and information have also exploded, and previous research methods have certain instability. Therefore, this study chooses to perform predictive analysis directly on stock data to avoid the complicated data acquisition process and the interference of redundant information and proposes a short-term stock forecasting method based on the SVR and RF Bagging ensemble algorithm. From a microscopic perspective, the data of the stocks in the Shanghai Stock Exchange Index from 2021 to 2022, including the trading date, opening index, highest index, lowest index, closing index, and trading volume, a total of 6 basic indicators, and based on this, 15 lagging indicators are defined. Then randomly divide and cut, take 30 days as a cycle as training data, use R^2 and MSE to evaluate the effectiveness of the model, and compare and verify with the conventional stock prediction model. The purpose of this paper is to propose a general short-term stock forecasting model, thereby making some recommendations for quantitative investing. It also further deepens the use of the integrated model.

2. Construction of base learner

2.1. SVR method

Given a set of data samples $\{(x_i, y_i), \dots, (x_l, y_l)\}$, where $x_i \in R^n$, $y_i \in R$, $i=1, 2, 3, \dots, l$. Find a function $f(x)$ on R^n so that $y=f(x)$ can be used to infer the value of y corresponding to any input x . The SVR algorithm requires minimizing a convex function and defining a loss function that ignores errors within a certain upper and lower range of the true value, this type of function is also known as the ε -insensitive loss function. The solution of the loss function is characterized by the minimization of the function, and the ε -insensitive loss function is used to ensure the existence of a global minimum solution and the optimization of the reliability generalization bound [8]. The schematic diagram of SVR is shown in Figure 1.

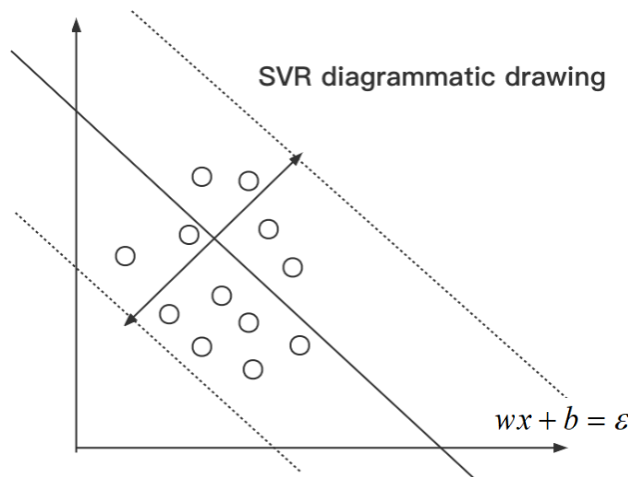


Figure 1. Schematic diagram of SVR

$$\min \frac{1}{2} \|w\|^2 \quad (1)$$

$$s.t. y_i (wx_i + b) \geq 1, \forall i \quad (2)$$

Using the Karush-Kuhn-Tucker (KKT) condition and the Lagrangian formula, the MMH can be expressed as the following "decision boundary".

$$d(X^T) = \sum_{i=1}^1 y_i \alpha_i X_i X^T + b_0 \quad (3)$$

$$f(x) = \sum_{i=1}^m (\hat{\alpha}_i - \alpha_i) k(x_i^T x) + b \quad (4)$$

Equation (3) represents the marginal maximally divided hyperplane. Equation 1 is the number of support vector points, because most of the points are not support vector points, and only individual points on the marginal hyperplane are support vector points. Then we sum up only the ones belonging to the support vector points; X_i is the eigenvalue of the support vector points; y_i is the category marker of the support vector points X_i , such as +1 or -1; X^T is the category attribute that can be calculated by bringing the test instance into this equation. α_i and b_0 are both single numerical parameters derived from the optimal algorithm mentioned above. α_i is the Lagrange multiplier. Whenever a new test sample X is available, it is brought into this equation and the magnitude of the equation value is calculated and classified. Equation (4) represents the expression of SVR, where $k(x_i^T x) = \varphi(x_i)^T \varphi(x_i)$ is the kernel function. The following kernel functions are usually selected, and the one with the best performance is chosen as the kernel function for a certain problem in this paper, a linear kernel function is selected for SVR, and the equation is as follows.

$$\kappa(x_i, x_j) = x_i^T x_j \quad (5)$$

2.2. RF method

Random Forest (RF) refers to a classifier that uses multiple trees to train and predict samples and is composed of multiple CART (Classification And Regression Tree). RF is to build a forest in a random way. There are many decision trees in the forest. There is no relationship between each decision tree in the random forest. After obtaining the forest, when a new input sample enters, let each decision tree in the forest make a judgment separately, and make predictions by voting on the category of the sample. Because the training of RF can be highly parallelized, it has advantages for the training speed of large samples in the era of big data. When the sample feature dimension is high, the model can still be trained efficiently. Compared with the Adaboost and GBDT of the Boosting series, the RF implementation is relatively simple, so this paper adopts RF as the base learner. The RF schematic diagram is shown in Figure 2 [9]:

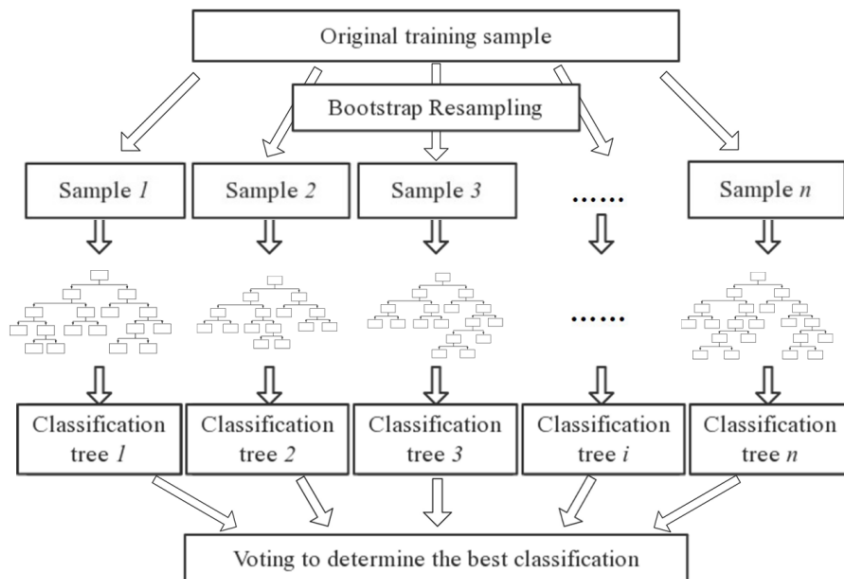


Figure 2. Schematic diagram of RF algorithm

The RF calculation steps are as follows:

STEP1: For the training set N , randomly extract N training samples from the training set as the training set of the tree;

STEP2: For the feature dimension of the sample is M , specify a constant $m \ll M$, randomly select m feature subsets from the M features, and select the best from the m features each time the tree is split;

STEP3: Make sure that each tree is growing to the maximum and that there is no pruning process.

3. Integrated Learning based on Bagging

Integrated learning (EL) is the process of building and combining multiple learners to accomplish a learning task. The general structure of EL is to generate a set of "individual learners" and then combine them with some strategy. The integration contains only individual learners of the same type, among which the individual learners are called "base learners" and the corresponding algorithm is called the "base learning algorithm". The integration contains different types of individual learners, among which the individual learners are called "group learners". Based on the generation method of individual learners, the integration learning methods are divided into two categories: serialized methods, which have strong dependencies among individual learners and must be generated serially, representing Boosting, and parallelized methods, which do not have strong dependencies among individual learners and can be generated simultaneously, representing Bagging. The simple voting method is used for classification and the simple averaging method is used for regression. If there is a shape congruence, either one is close.

The EL algorithm needs to train a Base Learner separately. For regression problems, the output of each base learner is averaged as the final result of the model. For the classification problem, according to the classification result of each base learner, the final Bagging result is obtained by voting, that is, the category with the most votes.

Specifically, each base learner is trained on the training set using a new dataset sampled by Bootstrap. The Bootstrap sampling process is as follows. Assuming that there are training samples, each time the base learner is trained, samples are randomly sampled with replacement. Repeat the process times if you want to train a base learner. Samples that are not used in the sampling process can be used as a test set to test the effect of final bagging. The schematic diagram of integrated learning is shown in Figure 3.

```
input: Training set  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ 
      Base learning algorithm  $\mathfrak{I}$ 
      the number of training rounds  $T$ 
Process:
1: for  $t=1, 2, \dots, T$  do
2:  $h_t = \mathfrak{I}(D, D_{bs})$ 
3: endfor
output:  $H(x) = \arg \max_{y \in Y} \sum_{t=1}^T \Pi(h_t(x) = y)$ 
```

Figure 3. Bagging integrated learning

4. Algorithm test index setting

Model building is very important, and model evaluation is also very important. Model evaluation is to judge whether the fitted model is excellent. In many cases, if the model is not tested, it is difficult to intuitively judge the correctness or accuracy of the model. Only by selecting an evaluation method that matches the problem can we quickly discover problems in model selection or training, and

optimize the model iteratively. Aiming at the algorithm prediction problem in this paper, this paper selects the R^2 and MSE indicators to quantify the algorithm accuracy and evaluate the algorithm model more completely [10].

Suppose the predicted value is $\hat{y} = \{\hat{y}_1, \hat{y}_2, \hat{y}_3, \dots, \hat{y}_n\}$, true value is $y = \{y_1, y_2, y_3, \dots, y_n\}$. The R^2 and MSE indicator formula can be organized into the following expressions.

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\bar{y} - y_i)^2} \quad (6)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (7)$$

Equation (6) represents R^2 (Decisive Factor) and is used to measure the fit of the model, the value range is $[0, 1]$, if R^2 turns out to be 0, it shows that the model fitting effect is very poor; If R^2 turns out to be 1, indicates that the model has no errors. The larger R^2 , the better the model fitting effect is. Equation (7) represents Mean Squared Error (MSE), which is used to evaluate the accuracy of the predicted value, and the value range is $[0, +\infty)$.

5. Case study

5.1. Object introduction and data collection

The Shanghai Stock Exchange (SSE) is the fastest growing emerging stock market in the world, and the SSE Composite Index (000001.SS) contains all stocks listed on the SSE, which is weighted by *share price × total equity*, reflecting the stock market's volatility. The volatility of the stock market is an inevitable market risk for investors, who can study the price changes of a single stock through fundamentals and technicals, while the price changes of multiple stocks need to be understood on a case-by-case basis, which increases the workload and difficulty. For investors, stock indices, as indicators of market price movements, provide market trends, and general trends, and can be used to test the effectiveness of investments. For academic and political researchers, it can also be used as a reference indicator to observe and forecast the socio-political and economic development situation.

In existing stock forecasting studies, stock trading data is regarded as an important factor that can give feedback on market information. In this paper, we download the data of all trading days of the SSE index from finance.yahoo.com for a total of 728 trading days from August 1, 2019 to August 1, 2022 for analysis, and the data include five items: trading date, opening index, highest index, lowest index, closing index, and volume Basic indicators.

5.2. Algorithmic Tuning

Fama [11] argues that "the basic assumption of all technical theories is that history tends to repeat itself, i.e., that past patterns of price behavior of individual securities will tend to repeat themselves in the future." Paiva et al. [12] used lagged return observation indicators and technical indicators for stock price forecasting. Therefore, in this paper, based on the underlying data collected, 15 lagged indicators and 7 technical indicators are defined and all indicators are uniformly normalized and mapped to the interval $[-1, 1]$. The selected calculated indicators are shown in Table 1.

Table 1. Calculation index

Attribute	Details	Attribute	Details
1	Date	12	$r_6 = \ln(\frac{\text{high price}_i}{\text{open price}_{i-1}})$
2	open price	13	$r_7 = \ln(\frac{\text{high price}_i}{\text{open price}_{i-2}})$
3	high price	14	$r_8 = \ln(\frac{\text{high price}_i}{\text{open price}_{i-3}})$
4	low price	15	$r_9 = \ln(\frac{\text{high price}_{i-1}}{\text{open price}_{i-1}})$
5	close price	16	$r_{10} = \ln(\frac{\text{high price}_{i-2}}{\text{open price}_{i-2}})$
6	volume	17	$r_{11} = \ln(\frac{\text{high price}_{i-3}}{\text{open price}_{i-3}})$
7	$r_1 = \ln(\frac{\text{close price}_i}{\text{close price}_{i-1}})$	18	$r_{12} = \ln(\frac{\text{low price}_i}{\text{open price}_i})$
8	$r_2 = \ln(\frac{\text{close price}_{i-1}}{\text{close price}_{i-2}})$	19	$r_{13} = \ln(\frac{\text{low price}_{i-1}}{\text{open price}_{i-1}})$
9	$r_3 = \ln(\frac{\text{close price}_{i-2}}{\text{close price}_{i-3}})$	20	$r_{14} = \ln(\frac{\text{low price}_{i-2}}{\text{open price}_{i-2}})$
10	$r_4 = \ln(\frac{\text{close price}_{i-3}}{\text{close price}_{i-4}})$	21	$r_{15} = \ln(\frac{\text{low price}_{i-3}}{\text{open price}_{i-3}})$
11	$r_5 = \ln(\frac{\text{high price}_i}{\text{open price}_i})$	22	close price _{next day}

This paper implements data processing and modeling based on the Python platform [13]. Three sets of 30 consecutive trading days are randomly intercepted as a period in three years of data, with dependent variable Y: next day closing index, and independent variables X: [opening index, highest index, lowest index, closing index, volume, and the remaining 15 lagging indicators]. Data Frame with Random. Randint is used in Python to implement random data interception. The time range of the random data set is shown in Table 2, and the interception range is shown in Figure 4.

Table 2. Random data set time range

Data1	Data2	Data3
2020-9-11	2022-4-22	2019-12-3
2020-10-30	2022-6-8	2020-1-14



Figure 4. Stock Data Interception Range

The problem in this paper is to use short-term data for short-term prediction, so the training set is set to 90% and the test set to 10%, and the model is trained using the training set and optimized by hyperparameters to obtain better generalization.

5.2.1 SVR

Different kernel types and parameter settings of SVR were tested on each dataset, and the Linear kernel function was finally selected with a mean coefficient of determination $R^2=0.992$ and mean square error $MSE=0.406$, which performed well and was the most stable, and the linear kernel SVR model will be used in the next integrated model in this paper.

5.2.2 RF

The hyperparameters that need to be selected in the random forest regression model are mainly: the number of decision trees, maximum depth of decision trees, the minimum number of separated samples, the minimum number of leaf node samples, k-fold cross-validation, and the trained model is random in nature and the results obtained each time are inconsistent. Randomized SearchCV+GridSearchCV was tried for random matching with grid matching, combining each hyperparameter and outputting the optimal combination, with about 1000 matches for each data set. The simulation results were average and the matching parameters took longer. Therefore, the random forest regression model was chosen to be fitted several times to view the model accuracy and the mean value recorded in Table.

5.2.3 Bagging

In the Bagging integrated regression model, random sampling is performed with a self-service sampling method, and each sampling set is used to train one base learner, and there is no dependency between base learners, which can be generated simultaneously, and the prediction results of n independent uncorrelated models are averaged. Bagging integration is performed based on the prediction results of the base learners, using the error function as the loss function, and voting learning for SVR and RF, based on the parallel relationship between the two.

5.2.4 BP neural network

Training time about 0.021s; activation function identity; solver lbfgs; learning rate 0.1; L2 regular term 1; number of iterations 1000; number of hidden layer 1 neurons 100.

5.2.5 XGBoost

The training time is about 0.296s; base learner gob tree; the number of base learners 100; learning rate 0.1; L1 regular term 0; L2 regular term 1; sample sign sampling rate 1; tree feature sampling rate 1; node feature sampling rate 1; the minimum weight of samples in leaf nodes 0; maximum depth of tree 10.

Based on the Bagging algorithm constructed in this paper, a simulation comparison with a total of four other algorithms, SVR, RF, BP, and XGBoost, on the test set was conducted, and the results are shown in Table 3.

Table 3. Comparison of the accuracy of each algorithm test set

	SVR		RF		EL		BP		XGBoost	
	R^2	MSE	R^2	MSE	R^2	MSE	R^2	MSE	R^2	MSE
Data1	0.994	0.010	0.913	0.152	0.999	0.001	-0.852	0.064	-17.977	0.627
Data2	0.845	0.022	0.582	0.060	0.991	0.001	0.146	0.053	-3.446	0.275
Data3	0.966	0.006	0.625	0.625	0.949	0.002	-4.086	0.101	-0.022	0.020
Avg.	0.935	0.013	0.707	0.279	0.980	0.001	-1.597	0.073	-7.148	0.307

As seen from the accuracy results in Table 3, the MSE is generally small due to the normalization of the data, where the SVR-RF-Bagging model has the highest accuracy and stability with an R^2 of 0.980 and an MSE of 0.001. The R^2 of the coefficient of determination of BP and XGBoost is less desirable on the test set, while the R^2 values of SVR and RF are in the top 3, with This is the reason why these two algorithms are chosen as the basic models of the integrated algorithms in this paper. The integrated algorithms based on Bagging have better robustness and generalization compared with several individual learner algorithms before integration, in which both R^2 and MES have certain

improvements compared with those before integration. This shows that the algorithm based on Bagging integration has a significant advantage in short-term stock price prediction.

6. Conclusion

In this study, we choose the method of short-term stock forecasting based on SVM and RF's Bagging integration algorithm, which defines 15 lagging indicators and 7 technical indicators from a micro perspective for the stock itself data in the SSE index from 2021 to 2022, and normalizes all indicators uniformly and maps them to the interval $[-1, 1]$, and uses R2 and MSE to evaluate the validity of the model with the conventional stock forecasting models for comparison and validation. It is concluded that the SVR-RF-Bagging model has the highest accuracy and stability with R2 of 0.980 and MSE of 0.001. the R2 of the coefficient of determination of BP and XGBoost is less satisfactory on the test set, while the R2 values of SVR and RF are in the top 3 with 0.935 and 0.707, respectively. So the algorithm based on Bagging after integration is comparable to Therefore, the integrated algorithm based on Bagging has better robustness and generalization compared with several individual learner algorithms before integration, in which R2 and MES have certain improvement compared with those before integration, so the integrated algorithm based on Bagging has obvious advantages in short-term stock price prediction.

References

- [1] Yu P, Yan X. Stock price prediction based on deep neural networks [J]. *Neural Computing and Applications*, 2020, 32(6): 1609-1628.
- [2] Bukhari A H, Raja M, Sulaiman M, et al. Fractional neuro-sequential ARFIMA-LSTM for financial market forecasting [J]. *IEEE Access*, 2020, PP(99): 1-1.
- [3] Ananthi M, Vijayakumar K. Stock market analysis using candlestick regression and market trend prediction (CKRM) [J]. *Journal of Ambient Intelligence and Humanized Computing*, 2020(8).
- [4] Nabipour M, Nayyeri P, Jabani H, et al. Deep learning for Stock Market Prediction [J]. 2020..
- [5] Watanabe N, Murata M, Ogawa T, et al. Exploration and Evaluation of Machine Learning-Based Models for Predicting Enzymatic Reactions [J]. *Journal of Chemical Information and Modeling*, 2020,
- [6] Sadorsky P, Mcleer M. A Random Forests Approach to Predicting Clean Energy Stock Prices [J]. *Journal of Risk and Financial Management*, 2021, 14.
- [7] Yang H, Huang K, King I, et al. Localized support vector regression for time series prediction [J]. *Neurocomputing*, 2009, 72(10-12): 2659-2669.
- [8] Li Xiaojun, Tang Pan. Stock price prediction based on technical analysis, fundamental analysis and deep learning [J]. *Statistics and Decision Making*, 2022, 38(02): 146-150. DOI: 10.13546/j.cnki.tjyjc.2022.02.029.
- [9] Zheng Yan, Cheng Wenming, Cheng Yue. Structural reliability analysis based on support vector machine regression [J]. *Mechanical Science and Technology*, 2011, 30(01): 52-56. DOI: 10.13433/j.cnki.1003-8728.2011.01.005.
- [10] Shao Jing, Chen Zuoning, Yin Hongwu. Information flow control model for Python and its implementation [J]. *Computer Application Research*, 2015, 32(10): 3065-3069.
- [11] Zhang Daheng, Zhang Yingjun, Zhang Chuang. Decision-making system of ship meteorological route based on BP neural network [J/OL]. *Chinese Ship Research*: 1-9 [2022-08-19]. DOI: 10.19693/j.issn.1673-3185.02565.
- [12] Gu Tianxia, Liu Qinming, Ye Chunming. Research on credit evaluation of supply chain finance based on BO-XGBoost and ensemble learning method [J/OL]. *Journal of Shanghai University of Science and Technology*: 1-8 [2022-08-19]. DOI: 10.13255/j.cnki.jusst.20211027005.
- [13] Dias P F, Nogueira C, Peixoto H G, et al. Decision-Making for Financial Trading: A Fusion Approach of Machine Learning and Portfolio Selection [J]. *Expert Systems with Applications*, 2018, 115: S0957417418305037-.