Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 114 (2017) 473–480

Procedia Computer Science

www.elsevier.com/locate/procedia

Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems, CAS October 30 – November 1, 2017, Chicago, Illinois, USA

A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters

Omer Berat Sezer[a,c], Murat Ozbayoglu[a1], Erdogan Dogdu[b]

[a]*TOBB University of Economics and Technology, Department of Computer Engineering, Ankara, 06560, Turkey*
[b]*Cankaya University, Department of Computer Engineering, Ankara, 06790, Turkey*
[c]*TUBITAK - Space Technologies Research Institute, Ankara, 06531, Turkey*

Abstract

In this study, we propose a stock trading system based on optimized technical analysis parameters for creating buy-sell points using genetic algorithms. The model is developed utilizing Apache Spark big data platform. The optimized parameters are then passed to a deep MLP neural network for buy-sell-hold predictions. Dow 30 stocks are chosen for model validation. Each Dow stock is trained separately using daily close prices between 1996-2016 and tested between 2007-2016. The results indicate that optimizing the technical indicator parameters not only enhances the stock trading performance but also provides a model that might be used as an alternative to Buy and Hold and other standard technical analysis models.

Peer-review under responsibility of the scientific committee of the Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems.

Keywords: Stock Trading; Stock Market; Deep Neural-Network; Evolutionary Algorithms; Technical Analysis;

1. Introduction

Computational Intelligence techniques have been used as part of stock trading systems for some time [1]. Neural networks are among one of the most popular choices. In some studies stock prices were directly used for time series forecasting, but in most cases, technical and/or fundamental analysis indicators were used as features for the neural network models [2-5]. Evolutionary algorithms, mostly genetic algorithms (GA) [6], have been used for constructing profitable trading systems [9,10], mostly for technical analysis optimization[8], or optimizing the neural network that is developed for stock trading [7]. The implementation of genetic algorithms and other evolutionary computation techniques used in financial studies are comprehensively covered in [13]. Meanwhile, combining evolutionary optimized technical analysis indicators as features for a neural network based stock trading model has not been studied extensively.

[1]* Corresponding author. Tel.:+90-312-4073; fax: +90-312-292-4180.
E-mail address: mozbayoglu@etu.edu.tr

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems.
10.1016/j.procs.2017.09.031

In this study we propose a model that combines genetic algorithms and neural networks together in a stock trading system in such a way that, the features that are provided to the neural network are the optimized technical analysis buy-sell trigger points. Our motivation for introducing such a model is to find the best possible distinctive technical analysis parameters as features for a profitable automated stock trading system.

2. Model Feature

In the literature, technical analysis indicators such as Moving Average Convergence and Divergence (MACD), Relative Strength Index (RSI), William % R, Simple Moving Average (SMA) are used to determine the trend direction in the market. In our proposed model, we use RSI for buy-sell points prediction and SMA for trend forecasting.

3.1 Relative Strength Index (RSI)

Relative Strength Index (RSI) is an oscillator type technical analysis indicator that shows the historical strength and weakness of stock prices. RSI values oscillate between 0 and 100. If the value is over 70, the stock is considered to be in the "overbought" region. Meanwhile, if the value is under 30, the stock is assumed to be in the "oversold" region. RSI value also compares losses and gains in a specified time period. RSI value is calculated as illustrated in equations (1) and (2).

$$RSI = 100 - 100 / (1+RS) \qquad (1)$$

$$RS = Average\ Gain / Average\ Loss \qquad (2)$$

3.2 Simple Moving Average (SMA)

Simple Moving Average (SMA) shows the moving average of the prices for a given period. Mostly, the intersection of the SMA values with different interval values are used to determine the trend direction. If 50-days SMA goes below the 200-days SMA value, it is assumed that the trend is down, and the prices will decrease relative to current prices. If 50-days SMA goes above the 200-days SMA value, then it is assumed that the trend is up, and the prices will increase.

3. Method

In this study, we propose a novel method that uses a genetic algorithm and deep Multilayer Perceptron (MLP). In our approach, we use RSI values of the stock prices to determine the buy and sell points for stocks. We also use Apache Spark and Spark MLlib library for big data analytics. As can be seen in Figure 1, our proposed method is divided into six main steps. We aim to determine the best fit for the buy, sell, and hold points in the time series of the corresponding stock prices. In our study, the daily stock prices of Dow 30 stocks are obtained from finance.yahoo.com for training and testing purposes. Stock prices between 1/1/1997 to 12/31/2006 are used as the training data and stock prices between 1/1/2007 to 1/1/2017 are used as the test data. In Phase 0, the downloaded stock prices are first normalized according to the adjusted close prices. In Phase 1, RSI values for different intervals (1 to 20 days) and SMA values for different intervals (50 and 200 days) are calculated using TA4J[2] (Technical Analysis For Java) library. SMA values are calculated to determine whether the trend is up or down.

In GA Phase (Genetic Algorithm) (Phase 2), the best RSI values for buy and sell points in downtrend and uptrend are found in the random initialized population. The best returned RSI chromosomes are accumulated in the list for the input of the MLP as the training data set.

[2] https://github.com/mdeverdelhan/ta4j

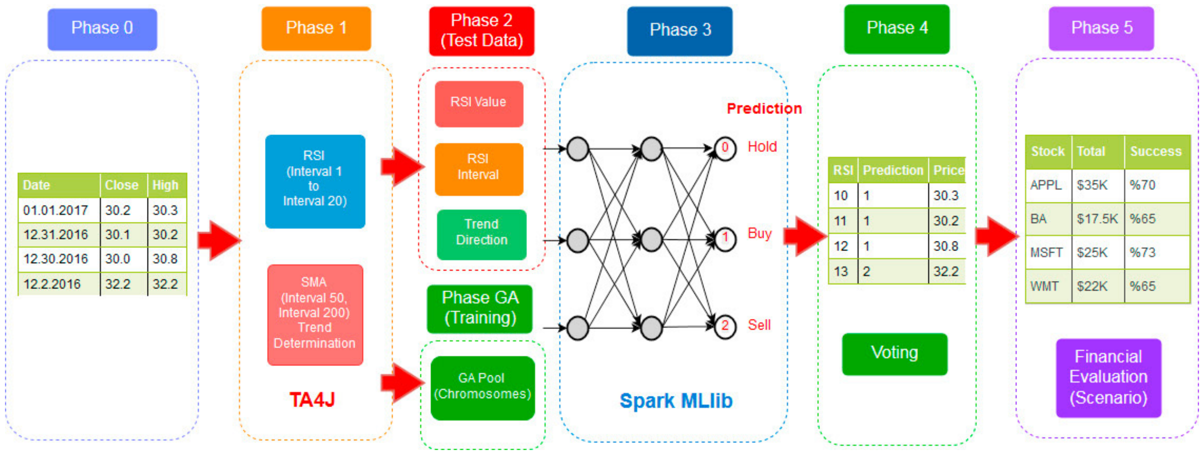

Fig. 1. Proposed Method (Genetic Algorithm and MLP)

Figure 3 illustrates the steps of the GA Phase. First, the chromosome population with 8 genes are randomly created. Figure 2 shows the structure of the chromosomes and genes in the chromosomes. The method to create the chromosomes in the population is as follows:

- RSI Buy values are created randomly between 5 and 40.
- RSI Buy intervals are created randomly between 5 and 20 days.
- RSI Sell values are created randomly between 60 and 95.
- RSI Sell intervals are created randomly between 5 and 20 days.
- The same procedure is followed to create 4 genes for uptrend.

Fig. 2. Chromosome with 8 genes

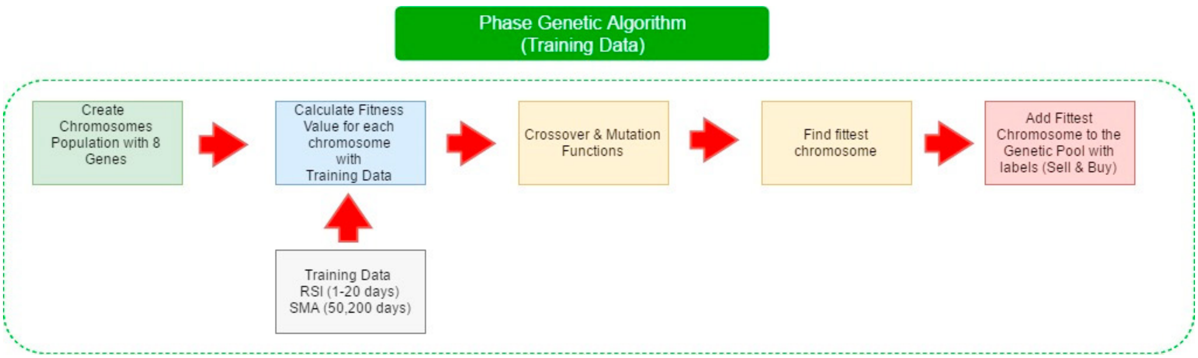

Fig. 3. Phase Genetic Algorithm

Initially, the number of the population of the chromosomes is 50. After initial population creation, fitness value for each chromosome is calculated using the training data, which includes RSI values with 1 to 20 days intervals and

SMA values with 50 and 200 days intervals. The fittest chromosome (most profitable chromosome) is found for the first generation. Then, the standard GA functions, crossover (0.7) and mutation (0.001) are applied on the chromosome population. After several generations, the fittest chromosome is found and it is added into the genetic pool list with labels (buy or sell) and trend direction (downtrend or uptrend). For instance, chromosome (26,10,62,8;29,6,79,15) is found as the fittest chromosome from population. Also, hold labeled values are added to the data pool with some representative in-between values as tabulated in Table 1.

Table 1. Training Data Example For MLP

Labels	RSI Value	RSI Interval (Days)	Trend Direction
1 (Buy)	26.0	10.0	0.0 (Downtrend)
2 (Sell)	62.0	8.0	0.0 (Downtrend)
1 (Buy)	29.0	6.0	1.0 (Uptrend)
2 (Sell)	79.0	15.0	1.0 (Uptrend)
0 (Hold)	50.0	18.0	0.0
0 (Hold)	45.0	7.0	0.0

As mentioned above, Phases 0, 1, and GA are applied on the training data (1/1/1997 to 12/31/2006). Then, Phases 0 and 1 are applied on the test data (1/1/2007 to 1/1/2017). In Phase 2, the test dataset is processed for the input of the MLP (RSI values, RSI intervals, and trend direction labels are calculated for the test dataset). In Phase 3, the training and test datasets are fed into the deep multilayer perceptron (MLP) using Apache Spark MLlib. MLP has 7 layers and its topology is configured as follows: (3, 20, 10, 8, 6, 5, 3). In the learning stage, it has 200 epochs for maximum iterations. The output of the MLP is a type of list that shows the prediction of the points at which the stock should be labeled as "sell", "buy", or "hold". The prediction list includes all possible RSI interval (1 to 20) predictions for all test datasets. In Phase 4, all possible RSI interval (1 to 20) predictions are reduced to one value

Table 2. Algorithm of Proposed Method

Algorithm: Predicting Label Points of Dow30 Stocks Using MLP + GA
procedure ALLPHASES()
Phase 0:
dataset = read (open, close, high, low, adjustedClose, volume)
dataset.adjustRatio = dataset.close / dataset.adjustedClose
adjust (dataset.open, dataset.close, dataset.high, dataset.low) with adjustRatio
Phase 1:
calculate RSI Values for 1,2,3,...,20 days intervals foreach line in dataset
trainingDataset, testDataset = dataset.split(dates=1997-2006, dates=2007-2016)
Phase GA:
create chromosomes population with 8 genes
calculate fitness value for each chrosome with trainingDataset
create fittest chromosome list with trainingDataset
Phase 2:
prepare testDataset for input of MLP
Phase 3:
model = MLP(layers=[3,20,10,8,6,5,3], maxEpochs=200, blocksize=128, seed=1234L)
model.train(trainingDataset)
model.test(testDataset)
Phase 4:
apply to voting mechanism for testDataset
Phase 5:
evaluateResults()

using a voting mechanism. The numbers of prediction results ("buy", "sell" and "hold") are counted. If the counter exceeds 14, that prediction is used as the actual prediction. Table 2 summarizes the algorithm of the proposed method. In addition, codes and resource files are available in [3].

Table 3. Genetic Algorithm + Deep MLP Dow 30 Results

Company	DMLP+GA	Rtn %	Ann.#ofT	PoS	Apt	L	MpT	MLT	MxC	MinC	IR
MMM	$17085.05	8.08%	5.5#	76.32%	1.68%	24#	22.64%	-11.16%	$18554.58	$8995.87	62.71%
AXP	$13908.78	4.9%	7.1#	67.35%	1.26%	23#	9.99%	-11.25%	$16208.65	$4046.03	55.08%
APPL	$37852.92	21.29%	7.1#	75.51%	3.32%	21#	9.67%	-12.42%	$39870.99	$7834.22	58.94%
BA	$23533.31	13.21%	5.4#	59.46%	3.33%	56#	25.5%	-12.88%	$24534.32	$4388.37	16.96%
CAT	$19394.75	10.08%	8.7#	61.67%	1.69%	18#	11.54%	-10.6%	-10.6%	$6002.75	55.76%
CVX	$31565.85	18.14%	5.9#	78.05%	3.12%	21#	22.17%	-11.67%	$31565.85	$9403.57	65.57%
CSCO	$20922.80	11.3%	6.5#	73.33%	2.03%	22#	12.36%	-11.95%	$20922.8	$7778.66	60.6%
KO	$26958.24	15.47%	6.1#	85.71%	2.48%	27#	9.63%	-11.53%	$26958.24	$9358.99	54.57%
DIS	$29958.82	17.25%	8.7#	68.33%	2.26%	29#	12.51%	-11.8%	$33943.53	$5171.4	30.06%
DD	$15304.38	6.37%	7.4#	66.67%	1.31%	17#	11.92%	-13.06%	$17146.9	$6584.94	64.73%
XOM	$22259.96	12.3%	9.4#	73.85%	1.39%	23#	13.42%	-10.45%	$22259.96	$10000.0	39.91%
GE	$19261.81	9.97%	4.5#	51.61%	3.91%	47#	104.33%	-10.83%	$19281.78	$3107.62	42.14%
GS	$16090.25	7.14%	7.3#	60.0%	1.89%	23#	56.84%	-12.06%	$19008.11	$4541.75	53.93%
HD	$19323.46	10.02%	6.1#	66.67%	2.05%	21#	8.51%	-10.81%	$20686.15	$8632.65	63.94%
IBM	$31134.92	17.9%	5.9#	75.61%	3.12%	26#	28.14%	-12.79%	$31483.38	$7977.44	56.99%
INTC	$17854.13	8.77%	6.5#	64.44%	1.74%	19#	8.3%	-11.95%	$19951.53	$7835.27	65.97%
JNJ	$19588.34	10.24%	7.0#	81.25%	1.51%	23#	11.73%	-12.5%	$19954.21	$9181.42	55.6%
JPM	$18136.83	9.02%	4.8#	57.58%	3.16%	70#	43.68%	-11.95%	$18136.83	$4577.41	7.19%
MCD	$25295.64	14.41%	6.1#	80.95%	2.4%	29#	7.19%	-10.83%	$26071.86	$10000.0	51.55%
MRK	$23467.98	13.17%	7.0#	72.92%	2.12%	21#	10.77%	-10.3%	$23467.98	$7337.6	58.42%
MSFT	$33313.59	19.07%	6.2#	76.74%	3.15%	26#	9.34%	-11.74%	$33313.59	$8933.32	54.13%
NKE	$40307.46	22.4%	6.5#	77.78%	3.5%	21#	17.89%	-11.24%	$49084.06	$9979.2	61.72%
PFE	$23219.50	12.99%	8.3#	71.93%	1.81%	21#	10.24%	-10.8%	$24496.08	$6611.99	51.95%
PG	$16713.78	7.32%	2.2#	80.0%	4.2%	150#	23.87%	-16.55%	$16713.78	$7554.79	10.29%
TRV	$22120.65	12.2%	4.4#	76.67%	2.95%	22#	23.51%	-12.47%	$22120.65	$9833.68	72.8%
UTX	$14407.36	5.44%	6.5#	73.33%	1.06%	19#	7.18%	-11.59%	$17906.82	$9495.03	65.45%
UNH	$19295.40	10.0%	5.2#	75.0%	2.07%	11#	9.16%	-8.35%	$19295.45	$5767.37	82.96%
VZ	$15455.29	6.52%	7.3#	70.0%	1.01%	20#	8.53%	-11.06%	$17572.32	$9585.46	59.13%
WMT	$20406.24	10.9%	6.1#	78.57%	1.84%	23#	8.53%	-11.73%	$21107.52	$10000.0	60.21%

4. Evaluation

After assigning actual prediction labels, the results are evaluated using the financial evaluation method in Phase 5. In our financial evaluation method, each stock is bought, sold or held according to the predicted label. Stock buy/sell transactions are applied as a real scenario. If the predicted label is "buy", the stock is bought at the current price. If the predicted label is "sell", the stock is sold at the current price. If the predicted label is "hold", no operation is performed at that point. Starting capital is $10,000 and each transaction is performed with all of the current available capital at that point. During trading, if the same label is repeated one after another in a sequence, only the first label gets triggered and the transaction is executed. Until the label changes, repeating labels are

[3] https://github.com/omerbsezer/SparkDeepMlpGADow30

ignored. In our financial evaluation, a near-real scenario is executed. For instance, trading commissions are applied ($1 per transaction) and stop loss situations (10%) are included.

For our model, we used the technical analysis parameter values that are associated with the buy-sell trigger points within the training set. As a result, the model attempts to learn "generally good entry and exit points" through technical analysis optimization from the training set. However, best trigger points do not necessarily match with the same values consistently on the technical indicators' outputs. Since we did not implement GA on the testing data, we were not able to tell what class (buy-sell-hold) would be associated with any particular data. Hence, we used

Table 4. Genetic Algorithm Dow 30 Results

Company	GA	Rtn %	Ann.#ofT	PoS	Apt	L	MpT	MLT	MxC	MinC	IR
MMM	$22136.47	12.21%	13.3#	72.83%	1.13%	11#	67.43%	-18.62%	$22482.69	$6583.49	58.34%
AXP	$17849.60	8.77%	8.1#	62.5%	2.98%	27#	191.68%	-15.33%	$24143.35	$3686.78	39.56%
APPL	$91402.33	37.83%	5.1#	74.29%	7.87%	56#	22.56%	-20.79%	$119272.44	$10000.0	21.92%
BA	$33257.97	19.04%	2.3#	68.75%	11.97%	82#	153.63%	-10.13%	$33257.97	$7005.03	47.70%
CAT	$20488.00	10.96%	4.4#	60.0%	4.21%	55#	36.73%	-25.19%	$31909.77	$8366.15	33.88%
CVX	$19312.81	10.01%	12.0#	62.65%	1.04%	17#	22.9%	-15.13%	$19313.21	$7882.83	41.38%
CSCO	$25693.83	14.67%	6.2#	74.42%	2.45%	20#	16.08%	-16.86%	$26053.04	$9627.76	64.77%
KO	$19330.74	10.03%	2.6#	94.44%	3.85%	35#	7.42%	-10.18%	$19330.74	$10000.0	74.46%
DIS	$22435.75	12.43%	6.4#	68.18%	2.26%	27#	28.32%	-13.09%	$23248.16	$7939.69	52.38%
DD	$22700.53	12.62%	7.8#	74.07%	1.81%	15#	12.47%	-15.01%	$22700.53	$8306.12	67.83%
XOM	$25710.99	14.68%	19.3#	64.66%	0.81%	8#	18.86%	-17.28%	$25710.99	$9862.75	52.62%
GE	$30446.40	17.52%	6.5#	64.44%	3.27%	31#	37.15%	-15.56%	$30501.84	$8144.14	43.84%
GS	$14806.92	5.86%	13.8#	65.26%	0.93%	15#	18.45%	-13.28%	$14806.92	$4316.05	42.53%
HD	$56318.78	28.49%	8.3#	71.93%	3.52%	31#	17.5%	-11.43%	$57164.64	$8697.29	29.63%
IBM	$29817.61	17.17%	3.2#	77.27%	5.67%	46#	45.68%	-11.13%	$29817.61	$10000.0	59.73%
INTC	$32661.97	18.73%	6.2#	69.77%	3.28%	36#	12.02%	-12.34%	$35200.68	$10000.0	37.85%
JNJ	$22428.83	12.43%	2.6#	88.89%	4.74%	58#	9.44%	-11.72%	$22428.83	$10000.0	58.10%
JPM	$34587.38	19.72%	16.1#	70.27%	1.61%	10#	79.62%	-15.16%	$34587.38	$9647.69	52.90%
MCD	$35502.32	20.17%	3.8#	76.92%	5.36%	61#	19.46%	-10.83%	$35502.32	$9983.75	36.34%
MRK	$23901.33	13.47%	5.2#	72.22%	2.91%	30#	10.77%	-11.68%	$23901.33	$6688.97	55.80%
MSFT	$32670.76	18.73%	3.5#	70.83%	5.82%	48#	16.41%	-17.65%	$32670.76	$9175.42	53.61%
NKE	$32914.74	18.86%	8.4#	74.14%	2.36%	23#	8.51%	-10.63%	$36727.31	$9595.73	45.63%
PFE	$24422.03	13.82%	4.6#	78.13%	3.16%	39#	20.3%	-21.13%	$24422.03	$8938.18	49.36%
PG	$16512.78	7.54%	26.1#	58.33%	0.36%	5#	28.5%	-15.23%	$16514.78	$7424.48	58.94%
TRV	$26815.77	15.38%	2.8#	89.47%	5.63%	83#	11.81%	-15.9%	$26815.77	$8608.88	36.70%
UTX	$28678.10	16.51%	11.5#	67.09%	1.55%	20#	8.84%	-12.12%	$32050.85	$7763.67	36.14%
UNH	$39884.30	22.22%	12.2#	59.52%	2.01%	19#	11.56%	-20.22%	$39891.12	$3851.79	35.98%
VZ	$20960.35	11.33%	13.1#	51.11%	0.91%	13#	14.37%	-11.06%	$22786.66	$8921.09	52.18%
WMT	$31000.88	17.83%	16.8#	73.28%	1.06%	13#	9.1%	-10.77%	$31000.88	$10000.0	37.13%

financial evaluation for our performance comparison instead of neural network performance to measure the effectiveness of our model. Our proposed method's (Genetic Algorithm and Deep Multi Layer Perceptron - GA+DMLP) evaluation results are shown in Table 3. Dow Jones 30 Stocks are evaluated using financial evaluation in terms of the following criteria (Table 3): the total capital with our proposed strategy (DMLP+GA), Our Annualized Return (Rtn%), Annualized Number of Transactions (Ann.#ofT), Percent of Success (PoS), Average Profit Percent Per Transactions (ApT), Average Transaction Length (L), Maximum Profit Percentage in Transaction

(MpT), Maximum Loss Percentage in Transaction (MLT), Maximum Capital (MxC), Minimum Capital (MinC), Idle Ratio (IR). Our proposed method's (DMLP+GA) average annualized return is 11.93%, and percent of success is 71.63%. We also evaluated another model using only Genetic Algorithm. The financial evaluation of GA is illustrated in Table 4. Proposed method's (GA) average annualized return is 15.83%, and the success rate is 70.88%. Table 5 shows the comparisons of our methods (GA+DMLP), (GA), (MLP), and "Buy and Hold" (BaH) strategy. We earlier used MLP with the indicators (RSI, MACD, William%R) [14][4]. [14]'s average annualized return is 10.3%, and the success ratio is 71.63%. Also, the average annualized return of BaH strategy is 13.83%. Generally, as mentioned in the literature, it is very difficult to beat Buy and Hold strategy during such a lengthy time period. Our first proposed strategy's (GA+MLP) annualized return performed better than BaH strategy's annualized return in 12 out of 29 stocks (Visa stock [V] did not have enough data points in the same period). Our second proposed strategy's (GA) annualized return performed better than BaH strategy's annualized return in 16 out of 29.

Table 5. Comparisons of MLP+GA, GA, MLP method [14], Buy & Hold Method (BAH)

Company	MLP+GA	MLP+GA Rtr	GA	GAnnRtr	Best Chromosome	MLP [14]	MLP Rtn[14]	BAH	BAH Rtn
MMM	$17085.05	8.08%	$22136.47	12.21%	30 19 81 19 25 5 61 5	$15234.16	6.33%	$29324.88	16.99%
AXP	$13908.78	4.9%	$17849.60	8.77%	21 7 78 16 34 9 65 12	$14727.15	5.80%	$15157.78	6.25%
APPL	$37852.92	21.29%	$91402.33	37.83%	14 9 98 6 7 2 87 14	$14742.93	5.83%	$104256.20	40.79%
BA	$23533.31	13.21%	$33257.97	19.04%	11 11 83 16 25 12 77 19	$17010.05	8.05%	$22809.31	12.78%
CAT	$19394.75	10.08%	$20488.00	10.96%	26 9 83 18 11 6 63 18	$10252.42	0.36%	$21030.51	11.44%
CVX	$31565.85	18.14%	$19312.81	10.01%	25 12 78 15 38 5 67 12	$17907.21	8.87%	$22968.13	12.89%
CSCO	$20922.80	11.3%	$25693.83	14.67%	16 19 81 10 22 6 71 12	$21182.93	11.57%	$13126.52	4.05%
KO	$26958.24	15.47%	$19330.74	10.03%	16 13 85 14 12 9 74 13	$17258.98	8.29%	$23354.41	13.18%
DIS	$29958.82	17.25%	$22435.75	12.43%	20 5 63 13 39 19 86 9	$28859.03	16.71%	$34368.91	19.72%
DD	$15304.38	6.37%	$22700.53	12.62%	29 5 68 11 6 6 75 7	$17750.91	8.74%	$22197.39	12.34%
XOM	$22259.96	12.3%	$25710.99	14.68%	32 5 67 7 35 5 60 6	$18385.49	9.30%	$15946.05	7.05%
GE	$19261.81	9.97%	$30446.40	17.52%	32 17 65 8 11 6 80 15	$12663.52	3.50%	$12399.64	3.19%
GS	$16090.00	7.14%	$14806.92	5.86%	38 6 69 5 29 7 83 11	$14230.22	5.28%	$12238.97	2.99%
HD	$19323.00	10.02%	$56318.78	28.49%	29 7 78 7 26 5 81 14	$15088.71	6.19%	$43768.70	24.04%
IBM	$31134.92	17.9%	$29817.61	17.17%	31 18 80 15 12 8 81 10	$17151.82	8.19%	$21143.52	11.55%
INTC	$17854.00	8.77%	$32661.97	18.73%	8 6 71 10 24 6 76 16	$27965.75	16.21%	$23656.29	13.40%
JNJ	$19588.34	10.24%	$22428.83	12.43%	21 9 79 17 15 9 76 18	$19043.10	9.86%	$23687.77	13.42%
JPM	$18136.83	9.02%	$34587.38	19.72%	14 5 70 13 31 5 63 5	$49181.78	26.17%	$22092.57	12.26%
MCD	$25295.64	14.41%	$35502.32	20.17%	21 14 93 8 14 5 82 12	$17519.35	8.53%	$38489.77	21.75%
MRK	$23467.98	13.17%	$23901.33	13.47%	14 7 71 11 31 16 71 18	$29081.32	16.86%	$18865.70	9.71%
MSFT	$33313.59	19.07%	$32670.76	18.73%	21 10 81 13 5 6 88 9	$37923.78	21.48%	$25820.00	14.85%
NKE	$40307.46	22.4%	$32914.74	18.86%	9 5 72 9 21 5 76 11	$22940.48	12.89%	$48496.06	25.93%
PFE	$23219.50	12.99%	$24422.03	13.82%	15 10 88 8 13 5 69 19	$11094.86	1.53%	$18953.47	9.78%
PG	$16713.78	7.32%	$16512.78	7.54%	39 7 76 15 33 5 84 1	$20278.23	10.88%	$17434.55	8.46%
TRV	$22120.65	12.2%	$26815.77	15.38%	31 13 76 15 28 14 92 8	$64371.78	31.23%	$31098.53	18.01%
UTX	$14407.36	5.44%	$28678.10	16.51%	39 5 91 6 30 6 67 11	$18540.16	9.43%	$20932.55	11.38%
UNH	$19295.40	10.0%	$39884.30	22.22%	34 9 84 13 36 6 65 15	$9343.90	-0.99%	$34464.65	19.80%
VZ	$15455.29	6.52%	$20960.35	11.33%	13 10 72 16 37 6 60 19	$12147.37	2.88%	$24315.17	13.83%
WMT	$20406.24	10.9%	$31000.88	17.83%	34 5 66 7 39 5 76 7	$32230.01	18.63%	$18389.92	9.30%

[4] https://github.com/omerbsezer/SparkMlpDow30

5. Conclusion

Utilizing optimized technical analysis feature parameter values as input features for neural network stock trading system is the basis for our proposed model. We used genetic algorithms to optimize RSI parameters for uptrend and downtrend market conditions. Then, we used those optimized feature values as buy-sell trigger points for our deep neural network data set. We used Dow 30 stocks to validate our model. The results indicate that such a trading system produces comparable or better results when compared with Buy & Hold and other trading systems for a wide range of stocks even for relatively longer periods. For future work, we plan on combining more technical parameters and utilize Convolutional Neural Networks (CNN) or other deep neural network models.

Acknowledgements

This paper is funded by TUBITAK (The Scientific and Technological Research Council of Turkey) through project grant no. 215E248.

References

1. Atsalakis, G. S., Valavanis, K.P. (2009) "Surveying stock market forecasting techniques – Part II: Soft computing methods", Expert Systems with Applications, vol 36, pp. 5932-5941.
2. Chen, A.S., Leung, M.T., Daouk, H. (2003) "Application of neural networks to an emerging financial market: forecasting and trading the Taiwan Stock Index", Computers & Operations Research, vol 30, pages: 901-923
3. Wang, J.Z., Wang, J.J., Zhang, Z.G., Guo, S.P. (2011) "Forecasting stock indices with back propagation neural network", Expert Systems with Applications, Volume 38, Issue 11, Pages 14346-14355
4. Guresen, E., Kayakutlu, G., Daim, T.U. (2011) "Using artificial neural network models in stock market index prediction", Expert Systems with Applications, Volume 38, Issue 8, Pages 10389-10397.
5. Liao, Z., Wang, J. (2010) "Forecasting model of global stock index by stochastic time effective neural network", Expert Systems with Applications, Volume 37, Issue 1, Pages 834-841
6. Holland, J. (1992) "Adaptation in Natural and Artificial Systems", MIT Press.
7. Kwon, Y.K., Moon, B.R. (2007) "A Hybrid Neurogenetic Approach for Stock Forecasting", IEEE Transactions on Neural Networks, Volume 18, No 3, pages 851-864.
8. Özbayoğlu, A.M. and Erkut, U. (2010), "Stock Market Technical Indicator Optimization by Genetic Algorithms", Artificial Neural Networks for Engineering 2010, pp. 589-596, St. Louis, MO, 1-3 November, 2010.
9. Ucar, İ., Ozbayoglu, A.M., Ucar, M., (2015) "Developing a Two Level Options Trading Strategy Based on Option Pair Optimization of Spread Strategies with Evolutionary Algorithms", 2015 IEEE Congress on Evolutionary Computation, 25-28 May 2015, Sendai, Japan
10. Ucar, M., Bayram, İ., Ozbayoglu, A.M. (2013) "A Two-level Cascade Evolutionary Computation based Covered Call Trading Model", Procedia Computer Science, Volume 20, Pages 472-477
11. Bisoi, R., Dash, P.K. (2014) "A hybrid evolutionary dynamic neural network for stock market trend analysis and prediction using unscented Kalman filter", Applied Soft Computing, vol 19,, pages 41-55
12. Evans, C., Pappas, K., Xhafa, F. (2013) "Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation", Mathematical and Computer Modelling, vol 58, pages 1249–1266
13. Aguilar-Rivera, R., Valenzuela-Rendon, M and Rodriguez-Ortiz, J.J.. (2015) "Genetic Algorithms and Darwinian approaches in financial applications: A survey.", Expert Systems with Applications, vol 42,. p. 7684-7697.
14. Sezer, O.B., Özbayoglu, A.M and Dogdu E. (2017) "An Artificial Neural Network-based Stock Trading System Using Technical Analysis and Big Data Framework", Proceedings of the SouthEast Conference. ACM, 2017. p. 223-226.