AITA: A new framework for Trading Forward Testing with an Artificial Intelligence Engine

Ivan Letteri[1,*]

[1]University of L'Aquila, via Vetoio snc, Coppito (AQ), 67100, Italy - Department of Information Engineering, Computer Science and Maths

Abstract

In general, traders test their trading strategies by applying them to historical market data (backtesting) and then reapply those that have made the most profit on that past data.

In this article, we propose AITA (*Artificial Intelligence Trading Assistant*), our framework for generating trading systems by applying the new trading strategy called DNN-forwardtesting [1] by determining the best strategy based on the prediction issued by a deep neural network. In this work, we show the experiment with AITA involves the use of 10 stocks that are first filtered according to their volatility using the *Kmean++* model. Having determined the assets with average volatility, we use this historical data to train a deep feed-forward neural network to predict price trends over the next 30 days of the open stock market. Finally, the trading system, created by AITA, calculates the most effective technical indicator by applying it to the DNN forecasts to generate the trading strategy.

The results confirm that neural networks outperform classical statistical techniques by increasing Sharpe, Sortino and Calmar ratios compared to even strategies chosen through traditional backtesting.

Keywords

Statistical Learning, Deep Learning, Multi Layer Perceptron, Kmeans, Technical Analysis, Stock Market Prediction, Trading System, Algorithmic Trading, Backtesting, Forwardtesting

1. Introduction

Stock market forecasting is considered a field of research with promising returns for investors. However, there are considerable challenges to forecasting trends accurately and sufficiently precisely due to their complexity, chaotic and non-linear nature. In fact, traditional statistical models, which have been widely applied to market trend forecasting so far, can easily handle only linear or stationary sequences.

In our experiment, we benchmarked the ARIMA (Autoregressive Integrated Moving Average) model, which is strong in its robustness and efficiency in terms of short-term forecasting [2] but showed poor results. On the other hand, artificial intelligence models are currently employed in a variety of tasks, e.g. to classify cyber-attacks [3], and predict network traffic anomalies [4, 5]. Among such artificial intelligence methods and, in particular, Deep Neural Networks (DNNs) have proven suitable for dealing with complex non-linear problems with multiple influencing factors [6].

In this paper, we propose AITA, our framework that uses historical stock price data to train a set of DNNs to predict future (next month) stock prices. These predictions are exploited in a new way to determine the most profitable technical indicators to be used as the basis of the trading strategy.

In this work, we show AITA (Artificial Intelligence Trading Assistant), our framework for generating trading systems which applies the new trading strategy called DNN-forwardtesting [1], instead of forward testing (also known *paper trading*) or backtesting. With our technique, the best strategy is devised by observing the profits earned by applying candidate strategies directly to the prediction of DNNs.

2. AITA features

2.1. Price Action notation

Technical analysis (TA) represents the type of investment analysis that uses simple mathematical formulations based on Price Action. TA uses the analysis of asset price history series [7], defined as OHLC, i.e., the opening, higher, lowest and closing prices of an asset, typically represented with candlesticks charts (see Fig. 1). For each timeframe t, the OHLC of an asset is represented as a 4-dimensional vector $X_t = (x_t^{(o)}, x_t^{(h)}, x_t^{(l)}, x_t^{(c)})^T$, where $x_t^{(l)} > 0$, $x_t^{(l)} < x_t^{(h)}$ and $x_t^{(o)}, x_t^{(c)} \in [x_t^{(l)}, x_t^{(h)}]$.

Ital-IA 2023: 3rd National Conference on Artificial Intelligence, organized by CINI, May 29–31, 2023, Pisa, Italy

*Corresponding author.
ivan.letteri@univaq.it (I. Letteri)
0000-0002-3843-386X (I. Letteri)

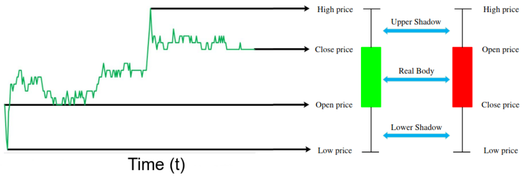

Figure 1: Example of a candlestick chart.

2.2. Volatility Estimators used

Volatility quantifies the dispersion of returns. Unfortunately, this dispersion can not be measured and volatility is not directly observable, but it is possible to estimate it [8]. AITA framework is designed to use the **Historical Volatility** measures[1].

- The *Parkinson* (PK) estimator incorporates the stock's daily *high* and *low* prices as follow: $PK = \sqrt{\frac{1}{4Nln2}\sum_{i=1}^{N}(ln\frac{x_t^{(h)}}{x_t^{(l)}})^2}$. It is derived from the assumption that the true volatility of the asset is proportional to the logarithm (ln) of the ratio of the high $x_t^{(h)}$ and low $x_t^{(l)}$ prices of N observations.

- The *Garman-Klass* (GK) estimator is calculated as follows: $GK = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(ln\frac{x_t^{(h)}}{x_t^{(l)}})^2 - \frac{1}{N}\sum_{i=1}^{N}(2ln(2)-1)(ln\frac{x_t^{(c)}}{x_t^{(o)}})^2}$. This method is robust for opening jumps in price and trend movements. However, the estimator assumes that price movements are log-normally distributed, which may not always be the case in practice.

- The *Rogers-Satchell* (RS) estimator uses the range of prices within a given time interval as a proxy for the volatility of the asset as follows: $RS = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(ln(\frac{x_t^{(h)}}{x_t^{(c)}})ln(\frac{x_t^{(h)}}{x_t^{(o)}}) + ln(\frac{x_t^{(l)}}{x_t^{(c)}})ln(\frac{x_t^{(l)}}{x_t^{(o)}})}$. RS assumes that the range of prices within the interval is a good proxy for the volatility of the asset, which may not always be the case. Additionally, the estimator may be sensitive to outliers and extreme price movements.

- The *Yang-Zhang* (YZ) estimator [9] incorporates OHLC prices as follows: $YZ = \sqrt{\sigma^2_{OvernightVol} + k\sigma^2_{OpentoCloseVol} + (1-k)\sigma^2_{RS}}$, where $k = \frac{0.34}{1.34+\frac{N+1}{N-1}}$, $\sigma^2_{Open2CloseVol} = \frac{1}{N-1}\sum_{i=1}^{N}(ln\frac{x_t^{(c)}}{x_t^{(o)}} - ln\frac{x_t^{(c)}}{x_t^{(o)}})^2$, and $\sigma^2_{OvernightVol} = \frac{1}{N-1}\sum_{i=1}^{N}(ln\frac{x_t^{(o)}}{x_{t-1}^{(c)}} - ln\frac{x_t^{(o)}}{x_{t-1}^{(c)}})^2$. Empirical studies have shown that the YZ estimator can perform well in a variety of settings, including in the presence of jumps and in the presence of non-normality in the data. However, like any estimator, it is not perfect and may have limitations in certain situations, and it is often recommended to use multiple estimators and compare their results to gain a more complete understanding of the underlying volatility.

2.3. Trading Strategies implemented

Two distinct trading strategy classes are used in AITA framework:

- *Trend Following.* One way to trade a trend is to look at an asset with a resistance line. Once the price breaks through resistance, a trader places an order in the direction of the breakout[2]. Trend-following, or momentum, strategies have the attractive property of generating trading returns with a positively skewed statistical distribution. Consequently, they tend to hold on to their profits and are unlikely to have severe 'drawdowns' ([10]).

- *Mean Reversion.* The idea of mean reversion strategies is that the maximum and minimum price of a security is temporary and that it will tend to average over time (see [11]). Elliott et al.[12] explain how mean-reverting processes might be used in pairs trading and developed several methods for parameter estimation.

It is worth noting that trend following and mean reversion strategies, although theoretically opposing ideas, are not in conflict with each other and they are therefore both applicable at the same time to the same security.

2.4. Metrics applied

Profit and risk metrics are crucial considerations in trading AITA framework evaluates the following, for the potential profitability of the investments and to manage the risk exposure.

(i) The *Maximum drawdown (MDD)* measures the largest decline from the peak in the whole trading period, to show the worst case, as follows: $MDD = max_{\tau\in(0,t)}[max_{t\in(0,\tau)}\frac{n_t-n_\tau}{n_t}]$. (ii) The *Sharpe ratio (SR)* is a risk-adjusted profit measure, which refers to the return per unit of deviation as follows: $SR = \frac{\mathbb{E}[r]}{[r]}$. (iii) The *Sortino ratio (SoR)* is a variant of the risk-adjusted profit measure, which applies DD as risk measure: $SoR = \frac{\mathbb{E}[r]}{DD}$. (iv) The *Calmar ratio (CR)* is another variant of the risk-adjusted profit measure, which applies MDD as risk measure: $CR = \frac{\mathbb{E}[r]}{MDD}$.

To check the goodness of trades, we mainly focused on the *Total Returns* $R_k(t)$ for each stock ($k = 1, ..., p$) in the time interval ($t = 1, ..., n$), where $TR = R_k(t) = \frac{Z_k(t+\Delta t)-Z_k(t)}{Z_k(t)}$, and furthermore analysing the standardized returns $r_k = (R_k - \mu_k)/\sigma_k$, with ($k = 1, ..., p$), where σ_k is the standard deviation of R_k, e μ_k denote the average overtime for the studied period.

[1] https://dynamiproject.files.wordpress.com/2016/01/measuring_historic_volatility.pdf

[2] https://en.wikipedia.org/wiki/Breakout_(technical_analysis)

3. Dataset Pre-processing

3.1. Volatility Stock Clustering

Table 1
List of 10 stocks randomly selected.

Ticker	Company	Market
CSGKF	Credit Suisse Group AG	Other OTC
EOG	*EOG Resources, Inc.*	*NYSE*
META	Meta Platforms, Inc.	Nasdaq GS
NKE	NIKE, Inc.	NYSE
DIS	Walt Disney Co.	NYSE
PG	Procter & Gamble Co.	NYSE
QQQ	Invesco QQQ Trust	Nasdaq GM
IBM	Business Machines Corp.	NYSE
ANF	*Abercrombie & Fitch Co.*	*NYSE*
CS	Credit Suisse Group AG	NYSE

From the ten stocks in tab. 1, we created a dataset composed by the time series of the PK, GK, RS, YZ historical volatility estimators. The data has been standardized and clustered with K-means++ [13] following this process:

1. Compute k-means++ clustering for different values of k. In our case, we varied k from 2 to 20 clusters.
2. For each k, is calculated the total within-cluster sum of squares (wss).
3. Plot the curve of wss according to the number of clusters k.
4. Find the location of a bend (knee) in the plot, which is generally considered as indicator of the appropriate number of clusters, and the best clustering in our experiments is for $k = 6$.

Fig. 2 clearly shows that, between the considered stocks, ANF and EOG have all the types of observations spread over all the k-means++ clusters for each historical Volatility estimator considered. This makes them good candidates for our final dataset because they gather all the characteristics (pros and cons) of the four history volatility estimators adopted.

3.2. Stocks Selected

ANF and EOG are certainly assets with a sometimes controversial trend and consequently well profitable if rightly analyzed. In order to prove that, the dataset is general enough to model a variety of different shares with more or less the same volatility coefficient (*vc*), AITA framework also proves that the price time series corresponding to the selected assets are completely uncorrelated, i.e., ANF and EOG do not influence each other. Therefore, we evaluated the *synchrony* between the two financial assets using (i) the *Pearson coefficient* [14] and the result is 0.28 (see Fig. 3), which confirms that the two stocks are almost completely uncorrelated. However, this is a measure of the *global synchrony* in the overall period.

(ii) *The Dynamic Time Warping* (DTW) algorithm calculates the optimal match between the two series by minimising the Euclidean distance between pairs of samples at the same time. The minimum path cost is $d = 209.95$, and such a large distance between the two stocks supports our hypothesis of a complete absence of influence between them.

4. Forecasting Methods

4.1. ARIMA model as benchmark

Fig. 4 shows the predictions on the closing prices made with such auto-selected optimal ARIMA model for $n = 30$ days following the training timespan, which corresponds to the 2507 days of the market from October 30, 2011, to October 16, 2021. Table 2 reports its relative (very high) error metrics of it.

Table 2
Error metrics of ARIMA on ANF and EOG stock price prediction.

	ARIMA				
	MSE	RMSE	MAE	MAPE	EVS
ANF	25.49	5.05	3.86	0.09	-0.02
EOG	56.23	7.50	5.42	0.06	-3.94

4.2. DNN model

In this section, we maintain the same forecasting objective of Section 4.1, i.e., $n = 30$ days following the training date. We empirically found that the neural network performs better is the Multi Layer Perceptron when its *input layer* is fed with the $t = 5$ previous values, i.e., the prices of the previous market week. In other words, to forecast the price of a day s, the input neurons will be presented to the prices of days $s - 1, \ldots, s - 5$, respectively. The network then outputs its price prediction via a single neuron in the output layer.

The resulting optimal geometry has two hidden layers composed of $10 * t$ and $5 * t$ neurons, respectively, as in [3] and [5]. In addition, to help reduce overfitting we applied a dropout of 20% on each of the two internal layers [15] and, to introduce non-linearity between layers, we used *ReLU* as the activation function. To estimate the network learning performance during the training we use the *L1loss* function, which measures the mean absolute error (MAE) between each predicted value and the corresponding real one. The optimisation algorithm used to minimise such loss function during the training is the *adaptive moment* (Adam).

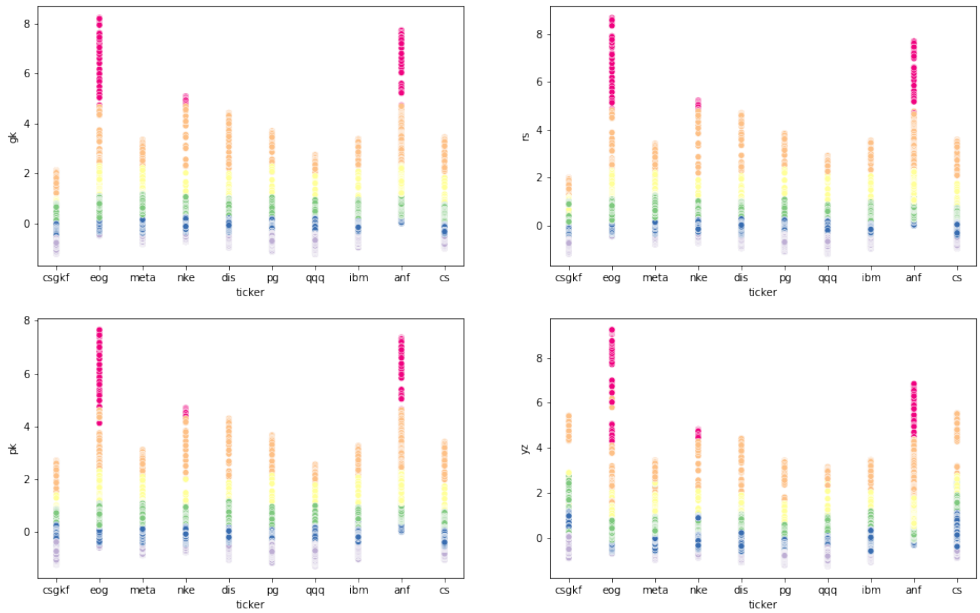

Figure 2: Kmeans++ Clusters with $k = 6$ of the Historical Volatility estimators dataset.

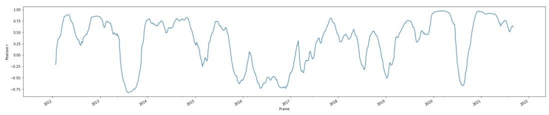

Figure 3: Pearson correlation between ANF and EOG.

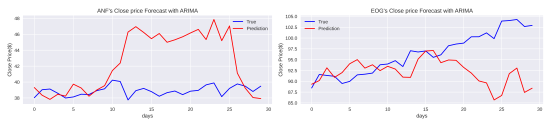

Figure 4: Detail of ARIMA forecast for the last 30 days of the ANF (left) and EOG (right) stock closing price.

Figure 5: Detail of DNN forecast for the last 30 days of the ANF (left) and EOG (right) stock closing prices.

Figure 5 shows the DNN forecasts on the ANF and EOG closing prices, respectively, in the same 30-day time frame used for the experiments of the previous section, whereas Table 3 reports the corresponding error metrics. It is clear that the DNN performs better than the statistical methods shown with ARIMA.

Table 3
Error metrics of DNN on ANF and EOG stock price prediction.

	DNN				
	MSE	**RMSE**	**MAE**	**MAPE**	**EVS**
ANF	1.75	1.32	1.07	0.02	0.91
EOG	2.39	1.55	1.23	0.01	0.7

5. The Experiment

After showing that DNNs are the best forecast technique for our stock prices dataset, we can introduce the novel trading system of AITA framework.

The AITA (algorithmic) trading strategy is encoded in a set of *entry* and *exit trading rules* which are in turn based on the value of a single indicator chosen from a set of twelve common technical indicators, i.e., Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), Bollinger Bands (BBs), Stochastics (ST), William %R (W%R), Momentum (MO), Relative Strength Index (RSI), Average True Range (ATR), Price Oscillator (PO) (see [16]), Triple Exponential Moving Average (TEMA, [17]) and Average Directional Index (ADX). We also tested some further meaningful combinations of the above indicators, like in [18], and [19], such as ST+MO+MACD, PO+W%R and PO+RSI.

AITA performed a *DNN-forwardtesting* of the strategies based on each of the above indicators on the 30-days

ANF	
Entry	$((x^{(l)} < TEMA^{(l)}) \vee (x^{(h)} < TEMA^{(h)})) \wedge ((x^{(c)} < TEMA^{(c)}) \vee (x^{(o)} < TEMA^{(o)}))$
Exit	$((x^{(l)} > TEMA^{(l)}) \vee (x^{(h)} > TEMA^{(h)})) \wedge ((x^{(c)} > TEMA^{(c)}) \vee (x^{(o)} > TEMA^{(o)}))$
EOG	
Entry	$(+DI > -DI) \wedge (ADX > 25)$
Exit	$(-DI > +DI) \wedge (ADX > 25)$

Figure 6: Trading system rules.

price forecasts following the training date ending on October 16th, 2021, generated by the DNNs developed in Section 4.2.

Our results show that the best indicator for ANF is the *Triple Exponential Moving Average*, whereas the *Average Directional Index* is more suitable for EOG. The corresponding trading rules, based on such indicators, are shown in Figure 6, where (o), (h), (l), (c) refer to the OHLC prices, respectively, and x is the current (opening, highest, etc.) price. Such rules were applied to the *possible future* during the forwardtesting.

Table 4
Performance of AITA forwardtesting-selected indicators.

	#Trades	TR ($)	ShR	SoR	CaR
ANF	3	6.126	2.194	3.340	12.403
EOG	3	1.374	1.253	2.556	5.814

Then, we evaluated the profit deriving from the application of such a strategy on the *real* data of the 30-day trading period following October 16th, 2021, having as starting point a budget of $100 invested in compound mode. The results are shown in Table 4.

As a baseline to compare such metrics, we re-evaluated the same set of technical indicators through the traditional backtesting technique on the historical data for the 30 days *before* October 16th, 2021, to see if it would result in different choices and maybe different profits. The results show that a trader using backtesting would choose ADX for the EOG share, as with our forward testing technique, so the profit would be the same in this case. However, the TEMA indicator would not be chosen for the ANF share. Indeed, the most promising indicator, given the past 30 days of the market, would be RSI (with an overbought 70 and oversell 30). However, if applied to the future, it would result in a *loss* of 1.16%, as shown in Table 5.

Table 5
Performance with the backtesting-selected indicator (RSI).

	#Trades	TR ($)	ShR	SoR	CaR
ANF	1	-1.168	0.119	0.158	-0.935

6. Conclusions

In this paper, we propose AITA framework, a stock market trading system that exploits deep neural networks as part of its main components, improving on previous work [20, 1].

The novelty of the approach implemented in AITA framework lies in the indicator selection technique that is completely different from the usual backtesting or real-time forward testing. AITA framework determines the most profitable indicator on the *probable future* predicted by a deep neural network trained on historical data.

As discussed, neural networks outperform the most common statistical methods in stock price forecasting predicting the future allowing for a very accurate selection of the indicator to be applied, which takes into account trends that would be very difficult to capture through backtesting.

To validate this claim, we applied our methodology to two very different assets with medium volatility, and the results show that our DNN-forwardtesting-based trading system achieves a profit equal to or higher than that of a traditional backtesting-based.

Given the promising potential of this approach, we will further test its reliability with more refined feature selection (e.g., [21, 4]) and balancing (buy, sell and hold trades) strategies (e.g., [22, 23]).

Finally, since such neural networks can be seen as black-box decision-making systems, we could also study the monitoring of machine ethics and the rules [24, 25, 26] related to their activity.

References

[1] I. Letteri, G. D. Penna, G. D. Gasperis, A. Dyoub, Dnn-forwardtesting: A new trading strategy validation using statistical timeseries analysis and deep neural networks, 2022. arXiv:2210.11532.

[2] A. A. Adebiyi, A. O. Adewumi, C. K. Ayo, Comparison of arima and artificial neural networks models for stock price prediction, Journal of Applied Mathematics 2014 (2014) 1–7. URL: https://EconPapers.repec.org/RePEc:hin:jnljam:614342.

[3] I. Letteri, G. D. Penna, G. D. Gasperis, Botnet detection in software defined networks by deep learn-

ing techniques 11161 (2018) 49–62. doi:10.1007/978-3-030-01689-0_4.

[4] I. Letteri, G. D. Penna, P. Caianiello, Feature selection strategies for HTTP botnet traffic detection (2019) 202–210. doi:10.1109/EuroSPW.2019.00029.

[5] I. Letteri, G. D. Penna, G. D. Gasperis, Security in the internet of things: botnet detection in software-defined networks by deep learning techniques, Int. J. High Perform. Comput. Netw. 15 (2019) 170–182. doi:10.1504/IJHPCN.2019.106095.

[6] I. Letteri, A. Di Cecco, G. Della Penna, New optimization approaches in malware traffic analysis, in: G. Nicosia, V. Ojha, E. La Malfa, G. La Malfa, G. Jansen, P. M. Pardalos, G. Giuffrida, R. Umeton (Eds.), Machine Learning, Optimization, and Data Science, Springer International Publishing, Cham, 2022, pp. 57–68. doi:10.1007/978-3-030-95467-3_4.

[7] H. Wang, W. Huang, S. Wang, Forecasting open-high-low-close data contained in candlestick chart, 2021. arXiv:2104.00581.

[8] G. Petneházi, J. Gáll, Exploring the predictability of range-based volatility estimators using rnns, 2018. URL: https://arxiv.org/abs/1803.07152. doi:10.48550/ARXIV.1803.07152.

[9] D. Yang, Q. Zhang, Drift-independent volatility estimation based on high, low, open, and close prices, The Journal of Business 73 (2000) 477–492. URL: http://www.jstor.org/stable/10.1086/209650.

[10] R. J. Martin, Design and analysis of momentum trading strategies (2021). URL: https://EconPapers.repec.org/RePEc:arx:papers:2101.01006.

[11] P. Huang, T. Wang, On the profitability of optimal mean reversion trading strategies, 2016. arXiv:1602.05858.

[12] R. J. Elliott, J. V. D. Hoek, W. P. Malcolm, Pairs trading, Quantitative Finance 5 (2005) 271–276. doi:10.1080/14697680500149370.

[13] D. Arthur, S. Vassilvitskii, K-means++: The advantages of careful seeding, volume 8, 2007, pp. 1027–1035. doi:10.1145/1283383.1283494.

[14] P. Wijayatunga, A geometric view on pearson's correlation coefficient and a generalization of it to non-linear dependencies, Ratio Mathematica 30 (2017). URL: https://doi.org/10.23755/rm.v30i1.5. doi:10.23755/rm.v30i1.5.

[15] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, CoRR abs/1207.0580 (2012). URL: http://arxiv.org/abs/1207.0580. arXiv:1207.0580.

[16] A. Barnwal, H. P. Bharti, A. Ali, V. Singh, Stacking with neural network for cryptocurrency investment, in: 2019 New York Scientific Data Summit (NYSDS), 2019, pp. 1–5. doi:10.1109/NYSDS.2019.8909804.

[17] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, A. Iosifidis, Forecasting stock prices from the limit order book using convolutional neural networks, in: 2017 IEEE 19th Conference on Business Informatics (CBI), volume 01, 2017, pp. 7–12. doi:10.1109/CBI.2017.23.

[18] P. S. K. Prasad, V. Madhav, R. Lal, V. Ravi, Optimal technical indicator-based trading strategies using nsga-ii, 2022. arXiv:2111.13364.

[19] A. Hryshko, T. Downs, System for foreign exchange trading using genetic algorithms and reinforcement learning, International Journal of Systems Science 35 (2004) 763–774. URL: https://doi.org/10.1080/00207720412331303697. doi:10.1080/00207720412331303697.

[20] I. Letteri, G. D. Penna, G. D. Gasperis, A. Dyoub, A stock trading system for a medium volatile asset using multi layer perceptron, 2022. arXiv:2201.12286.

[21] I. Letteri, G. D. Penna, L. D. Vita, M. T. Grifa, Mtakdd'19: A dataset for malware traffic detection 2597 (2020) 153–165. URL: http://ceur-ws.org/Vol-2597/paper-14.pdf.

[22] I. Letteri, A. D. Cecco, G. D. Penna, Dataset optimization strategies for malware traffic detection, CoRR abs/2009.11347 (2020). URL: https://arxiv.org/abs/2009.11347. arXiv:2009.11347.

[23] I. Letteri, A. D. Cecco, A. Dyoub, G. D. Penna, Imbalanced dataset optimization with new resampling techniques, in: K. Arai (Ed.), Intelligent Systems and Applications - Proceedings of the 2021 Intelligent Systems Conference, IntelliSys 2021, Amsterdam, The Netherlands, 2-3 September, 2021, Volume 2, volume 295 of *Lecture Notes in Networks and Systems*, Springer, 2021, pp. 199–215. doi:10.1007/978-3-030-82196-8_15.

[24] A. Dyoub, S. Costantini, F. A. Lisi, I. Letteri, Ethical monitoring and evaluation of dialogues with a mas 3002 (2021) 158–172.

[25] A. Dyoub, S. Costantini, I. Letteri, Care robots learning rules of ethical behavior under the supervision of an ethical teacher (short paper), volume 3281 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 1–8. URL: http://ceur-ws.org/Vol-3281/paper1.pdf.

[26] A. Dyoub, S. Costantini, I. Letteri, F. A. Lisi, A logic-based multi-agent system for ethical monitoring and evaluation of dialogues, in: A. Formisano, Y. A. Liu, B. Bogaerts, A. Brik, V. Dahl, C. Dodaro, P. Fodor, G. L. Pozzato, J. Vennekens, N. Zhou (Eds.), Proceedings 37th International Conference on Logic Programming (Technical Communications), ICLP Technical Communications 2021, Porto (virtual event), 20-27th September 2021, volume 345 of *EPTCS*, 2021, pp. 182–188. URL: https://doi.org/10.4204/EPTCS.345.32. doi:10.4204/EPTCS.345.32.