REFERENCES

[1] P. H. Vuong, L. H. Phu, T. H. Van Nguyen, L. N. Duy, P. T. Bao, and T. D. Trinh, "A bibliometric literature review of stock price forecasting: from statistical model to deep learning approach," *Science Progress,* vol. 107, no. 1, p. 00368504241236557, 2024.

[2] G. S. Atsalakis and K. P. Valavanis, "Surveying stock market forecasting techniques–Part II: Soft computing methods," *Expert Systems with Applications,* vol. 36, no. 3, pp. 5932-5941, 2009.

[3] R. Bisoi and P. K. Dash, "A hybrid evolutionary dynamic neural network for stock market trend analysis and prediction using unscented Kalman filter," *Applied Soft Computing,* vol. 19, pp. 41-56, 2014.

[4] S. Banerji, "Evolution of Stock Market in India: Historical Perspectives with Empirical Evaluations," in *The Indian Economy@ 75*: Routledge India, 2024, pp. 386-397.

[5] K. Shanu and A. Pandey, "Effect of Macroeconomic Variables on Stock Market Performance: A Study of Bombay Stock Exchange," in *Perspectives in Finance and Digital Transformations in Business*: Routledge India, 2024, pp. 11-19.

[6] S. Kulkarni, H. K. Pharasi, S. Vijayaraghavan, S. Kumar, A. Chakraborti, and A. Samal, "Investigation of Indian stock markets using topological data analysis and geometry-inspired network measures," *Physica A: Statistical Mechanics and its Applications,* vol. 643, p. 129785, 2024.

[7] S. Jana, "Stock market integration and trade: A study on India and its major trading partners," *Vision,* vol. 28, no. 3, pp. 313-326, 2024.

[8] Y.-J. Chen and Y.-M. Chen, "A fundamental analysis-based method for stock market forecasting," in *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP),* 2013: IEEE, pp. 354-359.

[9] R. D. Edwards, J. Magee, and W. C. Bassetti, *Technical analysis of stock trends*. CRC press, 2018.

[10] S. Sivanandam and S. Deepa, *Introduction to neural networks using Matlab 6.0*. Tata McGraw-Hill Education, 2006.

[11] M. Madhiarasan and S. Deepa, "A novel criterion to select hidden neuron numbers in improved back propagation networks for wind speed forecasting," *Applied intelligence,* vol. 44, no. 4, pp. 878-893, 2016.

[12] P. Shah *et al.*, "A comprehensive review on sentiment analysis of social/web media big data for stock market prediction," *International Journal of System Assurance Engineering and Management,* pp. 1-8, 2024.

[13] W. Dai, J.-Y. Wu, and C.-J. Lu, "Combining nonlinear independent component analysis and neural network for the prediction of Asian stock market indexes," *Expert systems with applications,* vol. 39, no. 4, pp. 4444-4452, 2012.

[14] C.-J. Lu, "Integrating independent component analysis-based denoising scheme with neural network for stock price prediction," *Expert Systems with Applications,* vol. 37, no. 10, pp. 7056-7064, 2010.

[15] S. Asadi, E. Hadavandi, F. Mehmanpazir, and M. M. Nakhostin, "Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction," *Knowledge-Based Systems,* vol. 35, pp. 245-258, 2012.

[16] M. H. Mohamed, "Rules extraction from constructively trained neural networks based on genetic algorithms," *Neurocomputing,* vol. 74, no. 17, pp. 3180-3192, 2011.

[17] J. Wang, S. Lu, S.-H. Wang, and Y.-D. Zhang, "A review on extreme learning machine," *Multimedia Tools and Applications*, vol. 81, no. 29, pp. 41611-41660, 2022.

[18] P. Vigneshvaran and A. V. Kathiravan, "Heart Disease Prediction using an optimized Extreme Learning Machine with Bacterial Colony optimization," in *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, 2022: IEEE, pp. 1425-1429.

[19] X. Tang and L. Chen, "Artificial bee colony optimization-based weighted extreme learning machine for imbalanced data learning," *Cluster Computing*, vol. 22, no. Suppl 3, pp. 6937-6952, 2019.

[20] J. Hu *et al.*, "Detection of COVID-19 severity using blood gas analysis parameters and Harris hawks optimized extreme learning machine," *Computers in Biology and Medicine,* vol. 142, p. 105166, 2022.

[21] S.-C. Ng, C.-C. Cheung, and S.-H. Leung, "Magnified gradient function with deterministic weight modification in adaptive learning," *IEEE Transactions on Neural Networks,* vol. 15, no. 6, pp. 1411-1423, 2004.

[22] D. Mohanty, A. K. Parida, and S. S. Khuntia, "Financial market prediction under deep learning framework using auto encoder and kernel extreme learning machine," *Applied Soft Computing,* vol. 99, p. 106898, 2021.

[23] S. Deng, Y. Zhu, Y. Yu, and X. Huang, "An integrated approach of ensemble learning methods for stock index prediction using investor sentiments," *Expert Systems with Applications,* vol. 238, p. 121710, 2024.

[24] X. Li *et al.*, "Empirical analysis: stock market prediction via extreme learning machine," *Neural Computing and Applications,* vol. 27, pp. 67-78, 2016.

[25] S. Das, T. P. Sahu, and R. R. Janghel, "Stock market forecasting using intrinsic time-scale decomposition in fusion with cluster based modified CSA optimized ELM," *Journal of King Saud University-Computer and Information Sciences,* vol. 34, no. 10, pp. 8777-8793, 2022.

[26] D. Wu, X. Wang, and S. Wu, "A hybrid method based on extreme learning machine and wavelet transform denoising for stock prediction," *Entropy,* vol. 23, no. 4, p. 440, 2021.

[27] A. Y. Fathi, I. A. El-Khodary, and M. Saafan, "A Hybrid Model Integrating Singular Spectrum Analysis and Backpropagation Neural Network for Stock Price Forecasting," *Rev. d'Intelligence Artif.*, vol. 35, no. 6, pp. 483-488, 2021.

[28] Y. Wu, D. Zhu, Z. Liu, and X. Li, "An improved BPNN algorithm based on deep learning technology to analyze the market risks of A+ H shares," *Journal of Global Information Management (JGIM),* vol. 30, no. 7, pp. 1-23, 2021.

[29] S. Das, T. P. Sahu, R. R. Janghel, and B. K. Sahu, "Effective forecasting of stock market price by using extreme learning machine optimized by PSO-based group oriented crow search algorithm," *Neural Computing and Applications,* vol. 34, no. 1, pp. 555-591, 2022.

[30] F. Zhang, "Extreme learning machine for stock price prediction," *International Journal of Electrical Engineering Education,* 2021.

[31] L. Xiong and Y. Lu, "Hybrid ARIMA-BPNN model for time series prediction of the Chinese stock market," in *2017 3rd International conference on information management (ICIM)*, 2017: IEEE, pp. 93-97.

[32] M. Vochozka, J. Horak, and T. Krulicky, "Innovations in management forecast: Time development of stock prices with neural networks," *Marketing & Management of Innovations,* no. 2, 2020.

[33] H. Hu, L. Tang, S. Zhang, and H. Wang, "Predicting the direction of stock markets using optimized neural networks with Google Trends," *Neurocomputing,* vol. 285, pp. 188-195, 2018.

[34] D. Jia and R. Xue, "Research on earnings management of growth enterprise market in China Stock Market: comparative analysis based on the BPNN, GBDT, and MLR models," *Computational Intelligence and Neuroscience,* vol. 2022, 2022.

[35] M. Jiang, L. Jia, Z. Chen, and W. Chen, "The two-stage machine learning ensemble models for stock price prediction by combining mode decomposition, extreme learning machine and improved harmony search algorithm," *Annals of Operations Research,* pp. 1-33, 2022.

[36] S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia, and D. C. Anastasiu, "Stock price prediction using news sentiment analysis," in *2019 IEEE fifth international conference on big data computing service and applications (BigDataService),* 2019: IEEE, pp. 205-208.

[37] N. Jing, Z. Wu, and H. Wang, "A hybrid model integrating deep learning with investor sentiment analysis for stock price prediction," *Expert Systems with Applications,* vol. 178, p. 115019, 2021.

[38] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM," *Neural Computing and Applications,* vol. 32, pp. 9713-9729, 2020.

[39] R. Gupta and M. Chen, "Sentiment analysis for stock price prediction," in *2020 IEEE conference on multimedia information processing and retrieval (MIPR),* 2020: IEEE, pp. 213-218.

[40] S. Wu, Y. Liu, Z. Zou, and T.-H. Weng, "S_I_LSTM: stock price prediction based on multiple data sources and sentiment analysis," *Connection Science,* vol. 34, no. 1, pp. 44-62, 2022.

[41] P. Mehta, S. Pandya, and K. Kotecha, "Harvesting social media sentiment analysis to enhance stock market prediction using deep learning," *PeerJ Computer Science,* vol. 7, p. e476, 2021.

[42] Y. Mehta, A. Malhar, and R. Shankarmani, "Stock price prediction using machine learning and sentiment analysis," in *2021 2nd International Conference for Emerging Technology (INCET),* 2021: IEEE, pp. 1-4.

[43] N. Darapaneni *et al.,* "Stock price prediction using sentiment analysis and deep learning for Indian markets," *arXiv preprint arXiv:2204.05783,* 2022.

[44] K. Srijiranon, Y. Lertratanakham, and T. Tanantong, "A hybrid Framework Using PCA, EMD and LSTM methods for stock market price prediction with sentiment analysis," *Applied Sciences,* vol. 12, no. 21, p. 10823, 2022.

[45] R. Batra and S. M. Daudpota, "Integrating StockTwits with sentiment analysis for better prediction of stock price movement," in *2018 international conference on computing, mathematics and engineering technologies (ICoMET)*, 2018: IEEE, pp. 1-5.

[46] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "Predicting Stock Market Price Movement Using Sentiment Analysis: Evidence From Ghana," *Appl. Comput. Syst.,* vol. 25, no. 1, pp. 33-42, 2020.

[47] Y. Guo, "Stock price prediction based on LSTM neural network: the effectiveness of news sentiment analysis," in *2020 2nd International Conference on Economic Management and Model Engineering (ICEMME)*, 2020: IEEE, pp. 1018-1024.

[48] A. Derakhshan and H. Beigy, "Sentiment analysis on stock social media for stock price movement prediction," *Engineering Applications of Artificial Intelligence,* vol. 85, pp. 569-578, 2019.

[49] A. Pathak and N. P. Shetty, "Indian stock market prediction using machine learning and sentiment analysis," in *Computational Intelligence in Data Mining: Proceedings of the International Conference on CIDM 2017*, 2019: Springer, pp. 595-603.

[50] M. Kim, E. L. Park, and S. Cho, "Stock price prediction through sentiment analysis of corporate disclosures using distributed representation," *Intelligent Data Analysis,* vol. 22, no. 6, pp. 1395-1413, 2018.

[51] S. Albahli, A. Irtaza, T. Nazir, A. Mehmood, A. Alkhalifah, and W. Albattah, "A Machine Learning Method for Prediction of Stock Market Using Real-Time Twitter Data," *Electronics*, vol. 11, no. 20, p. 3414, 2022.

[52] R. Kumar, S. Srivastava, A. Dass, and S. Srivastava, "A novel approach to predict stock market price using radial basis function network," *International Journal of Information Technology,* vol. 13, pp. 2277-2285, 2021.

[53] K. A. Rashedi, M. T. Ismail, N. N. Hamadneh, S. A. Wadi, J. J. Jaber, and M. Tahir, "Application of radial basis function neural network coupling particle swarm optimization algorithm to classification of Saudi Arabia stock returns," *Journal of Mathematics,* vol. 2021, pp. 1-8, 2021.

[54] V. Saravanan, L. Paudel, P. Acharya, P. Paramasivam, and A. S. Pillai, "An Efficient LSTM-Based Deep Learning Model for Stock Prediction Analytics and Real-time Visualization," 2023.

[55] C. Manjunath, B. Marimuthu, and B. Ghosh, "Analysis of Nifty 50 index stock market trends using hybrid machine learning model in quantum finance," *International*

Journal of Electrical and Computer Engineering (IJECE), vol. 13, no. 3, pp. 3549-3560, 2023.

[56] P. R. Mohapatra, A. K. Parida, S. K. Swain, and S. S. Basa, "Gradient Boosting and LSTM Based Hybrid Ensemble Learning for Two Step Prediction of Stock Market," *Journal of Advances in Information Technology,* vol. 14, no. 6, 2023.

[57] C. M. Liapis, A. Karanikola, and S. Kotsiantis, "Investigating deep stock market forecasting with sentiment analysis," *Entropy,* vol. 25, no. 2, p. 219, 2023.

[58] A. Rodrigues, S. Salve, T. Shaikh, and P. Bhilare, "Predicting Stock Market Prices with Sentiment Analysis and Ensemble Learning Techniques: A Hybrid Approach," 2023.

[59] P. Jishtu, H. Prajapati, and J. Fiaidhi, "Prediction of the stock market based on machine learning and sentiment analysis," *Authorea Preprints,* 2023.

[60] J.-Y. Huang, C.-L. Tung, and W.-Z. Lin, "Using social network sentiment analysis and genetic algorithm to improve the stock prediction accuracy of the deep learning-based approach," *International Journal of Computational Intelligence Systems,* vol. 16, no. 1, p. 93, 2023.

[61] N. Zaman *et al.*, "Stock market prediction based on machine learning and social sentiment analysis," *Authorea Preprints,* 2023.

[62] P. K. Mallick, A. Ranjan Panda, A. K. Parida, M. Ranjan Panda, and S. Rani Samanta, "Stock market prediction using hybrid multi-layer decomposition and optimized multi-kernel extreme learning machine," *Scientia Iranica,* vol. 30, no. 5, pp. 1625-1644, 2023.

[63] P. Rajendiran and P. Priyadarsini, "Survival study on stock market prediction techniques using sentimental analysis," *Materials Today: Proceedings,* vol. 80, pp. 3229-3234, 2023.

[64] https://finance.yahoo.com/.

[65] M. Lu and X. Xu, "TRNN: An efficient time-series recurrent neural network for stock price prediction," *Information Sciences,* vol. 657, p. 119951, 2024.

[66] F. Zhang, "Extreme learning machine for stock price prediction," *The International Journal of Electrical Engineering & Education,* p. 0020720920984675, 2021.

[67] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing,* vol. 70, no. 1-3, pp. 489-501, 2006.

[68] C. Wang and D. J. Hill, *Deterministic learning theory for identification, recognition, and control*. CRC Press, 2009.

[69] N. Sin-Chun and S.-H. L. Chi-Cheung, "Magnified gradient function with deterministic weight modification in adaptive learning," *Neural Networks,* vol. 6, pp. 1411-1423, 2004.

[70] K. Velusamy and R. Amalraj, "Cascade correlation neural network with deterministic weight modification for predicting stock market price," *IOP Conference Series: Materials Science and Engineering,* vol. 1110, no. 1, p. 012005, 2021/03/01 2021, doi: 10.1088/1757-899x/1110/1/012005.

[71] H. Li, "Support vector machine," in *Machine Learning Methods*: Springer, 2023, pp. 127-177.

[72] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458,* 2015.

[73] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern recognition,* vol. 77, pp. 354-377, 2018.

[74] W. Antweiler and M. Z. Frank, "Is all that talk just noise? The information content of internet stock message boards," *The Journal of finance,* vol. 59, no. 3, pp. 1259-1294, 2004.

[75] M. Jaggi, P. Mandal, S. Narang, U. Naseem, and M. Khushi, "Text mining of stocktwits data for predicting stock prices," *Applied System Innovation,* vol. 4, no. 1, p. 13, 2021.

[76] F. Wang, Y. Zhang, Q. Rao, K. Li, and H. Zhang, "Exploring mutual information-based sentimental analysis with kernel-based extreme learning machine for stock prediction," *soft computing,* vol. 21, pp. 3193-3205, 2017.

[77] F. Ecer, S. Ardabili, S. S. Band, and A. Mosavi, "Training multilayer perceptron with genetic algorithms and particle swarm optimization for modeling stock price index prediction," *Entropy,* vol. 22, no. 11, p. 1239, 2020.

[78] W. Khan, M. A. Ghazanfar, M. A. Azam, A. Karami, K. H. Alyoubi, and A. S. Alfakeeh, "Stock market prediction using machine learning classifiers and social media, news," *Journal of Ambient Intelligence and Humanized Computing,* pp. 1-24, 2022.

APPENDIX – I

Table 37 : HDFC dataset

S.NO	Date	Open	High	Low	Close	Adj Close	Volume
1.	10/07/2019	1189.38	1199	1183.13	1194.38	1159.65	5784574
2.	11/07/2019	1201.5	1207	1197.82	1203.57	1168.58	4548798
3.	12/07/2019	1209	1209	1192.53	1196.95	1162.15	4488136
4.	15/07/2019	1198	1201.5	1188.68	1197.38	1162.56	4116592
5.	16/07/2019	1209	1209	1191	1195.6	1160.84	6385284
6.	17/07/2019	1196.47	1205.8	1195.65	1198.72	1163.87	3879868
7.	18/07/2019	1201.5	1214.5	1200.95	1205.95	1170.89	5401292
8.	19/07/2019	1207.5	1210.35	1183.5	1187.82	1153.29	4469614
9.	22/07/2019	1172.5	1174.93	1140.63	1148.63	1115.23	11085648
10.	23/07/2019	1147.25	1147.25	1121.03	1131.75	1098.85	13901432
11.	24/07/2019	1128.38	1146.5	1125.5	1140.45	1107.29	8681870
12.	25/07/2019	1142.5	1157.72	1140.13	1143.03	1109.79	9731004
13.	26/07/2019	1139.1	1146.82	1136.43	1138.65	1105.55	6414914
14.	29/07/2019	1136.78	1136.78	1111.53	1122.15	1089.53	4988246
15.	30/07/2019	1127.5	1135.6	1119.95	1126.13	1093.38	5197068
16.	31/07/2019	1120	1133.5	1117.13	1125.82	1093.09	6945920
17.	01/08/2019	1114.55	1120	1098.5	1110.9	1081	9120838
18.	02/08/2019	1105.5	1112	1091	1107.18	1077.38	13912768
19.	05/08/2019	1098.53	1100	1081.25	1089.63	1060.3	9911220
20.	06/08/2019	1083.5	1105.13	1083.5	1094.55	1065.09	12445470

21.	07/08/2019	1095	1101	1082.5	1092	1062.61	9375068
22.	08/08/2019	1099	1121.5	1091.5	1116.57	1086.53	9855800
23.	09/08/2019	1123.5	1144.5	1119.9	1141	1110.29	15218056
24.	13/08/2019	1136.65	1136.65	1101.3	1110.18	1080.3	7720034
25.	14/08/2019	1117.7	1117.7	1101.53	1114.72	1084.72	7127026
26.	16/08/2019	1107	1116.5	1102.05	1113.85	1083.87	7623780
27.	19/08/2019	1119.25	1121	1100.85	1103.43	1073.73	4249100
28.	20/08/2019	1104.63	1113.9	1103.43	1110.3	1080.42	5135296
29.	21/08/2019	1112.45	1120.6	1107.1	1112.93	1082.97	5236356
30.	22/08/2019	1110.43	1110.43	1083.53	1087.05	1057.79	6996022
31.	23/08/2019	1080.5	1088.5	1069.8	1081.35	1052.25	10049808
32.	26/08/2019	1097.45	1133.07	1083.5	1128.07	1097.72	12640122
33.	27/08/2019	1128	1136.45	1120.05	1129.97	1099.56	10330416
34.	28/08/2019	1125	1128.68	1113.18	1123.75	1093.51	6831294
35.	29/08/2019	1123	1126.85	1110.15	1113.47	1083.51	7391858
36.	30/08/2019	1116.75	1130	1111	1113.97	1084	9616660
37.	03/09/2019	1110	1110.5	1098.07	1105.47	1075.72	5714338
38.	04/09/2019	1103.5	1137	1099.5	1123.88	1093.63	7530352
39.	05/09/2019	1124.5	1136.2	1112.5	1117.57	1087.5	8648126
40.	06/09/2019	1114	1128.3	1110.65	1122.95	1092.73	6435002
41.	09/09/2019	1123	1130.45	1108.05	1124.8	1094.53	5321702
42.	11/09/2019	1137	1137	1120.57	1125.65	1095.36	7604558
43.	12/09/2019	1125.35	1144.4	1124.07	1135.43	1104.87	9066150

44.	13/09/2019	1125.5	1133.72	1119	1128.72	1098.35	7685670
45.	16/09/2019	1125.95	1131.32	1120.5	1122.07	1091.88	5515092
46.	17/09/2019	1123.95	1124.25	1099.05	1105.68	1075.92	5328278
47.	18/09/2019	1108.65	1112.07	1090	1093.88	1064.44	6479942
48.	19/09/2019	1099.9	1107.05	1084	1101.05	1071.42	5311655
49.	20/09/2019	1108	1209.9	1105.4	1199.6	1167.32	23075017
50.	23/09/2019	1259.55	1282.7	1229	1257.25	1223.41	20960205

Table 38 : ICICI bank dataset

S.No	Date	Open	High	Low	Close	Adj Close	Volume
1.	21/12/2015	227.36	235.91	227.27	234.73	221.58	14307700
2.	22/12/2015	233.18	239.55	233.14	235.95	222.73	21342967
3.	23/12/2015	236	239.05	235.55	238.05	224.71	11533960
4.	24/12/2015	239.09	239.64	233.91	234.5	221.36	6097366
5.	28/12/2015	236.41	240.91	236.36	240.05	226.6	12041784
6.	29/12/2015	240	242.27	237.73	240.68	227.2	9054719
7.	30/12/2015	239.32	240.41	237.95	238.5	225.14	10025565
8.	31/12/2015	238.27	239.86	236.5	237.59	224.28	8546444
9.	01/01/2016	237.55	239.64	234.55	239.09	225.69	5998096
10.	04/01/2016	237.27	237.59	231.5	232.32	219.3	9435792
11.	05/01/2016	232.95	234.09	228.82	233.36	220.29	8966977
12.	06/01/2016	232.18	233.41	226.55	227.36	214.62	17416181
13.	07/01/2016	224	225.18	221	224.32	211.75	18240712
14.	08/01/2016	222.27	225.27	221.32	222.77	210.29	14404391
15.	11/01/2016	219.91	221.73	216.41	217.68	205.48	14208668
16.	12/01/2016	219.09	219.86	214.18	215.23	203.17	11895899
17.	13/01/2016	216.18	220.82	210.09	217.73	205.53	18215710
18.	14/01/2016	212.91	218	210.73	214.32	202.31	14600781
19.	15/01/2016	214.18	214.82	201.59	204.05	192.61	22333902
20.	18/01/2016	203.82	207.91	198	202.82	191.45	30239009
21.	19/01/2016	204.23	210.23	204.23	207.95	196.3	16222439

22.	20/01/2016	204.45	207.18	200.09	204	192.57	14353478
23.	21/01/2016	207.73	212.27	203.82	205.73	194.2	15665863
24.	22/01/2016	211.45	213.55	208.91	211.59	199.74	15148522
25.	25/01/2016	214.5	216.82	212.91	214.59	202.57	17137268
26.	27/01/2016	216.36	217.55	213.36	215.73	203.64	12155561
27.	28/01/2016	216.86	217.77	210.23	212	200.12	24258166
28.	29/01/2016	199.09	213	199.09	209.23	197.5	47755494
29.	01/02/2016	206.36	206.68	196.27	197.45	186.39	40395715
30.	02/02/2016	198.18	199.73	190.64	191.27	180.56	31821419
31.	03/02/2016	188.18	189.82	184.14	185.5	175.11	23592288
32.	04/02/2016	187.64	187.64	182.27	185.41	175.02	30869158
33.	05/02/2016	186.82	191.82	185.09	190.36	179.7	36189377
34.	08/02/2016	191.82	195.91	187.41	189.59	178.97	19366081
35.	09/02/2016	185.45	192.27	182.5	190.36	179.7	20906682
36.	10/02/2016	184.64	190.45	183.64	188.41	177.85	22742128
37.	11/02/2016	185.23	188.14	180	181.14	170.99	31817116
38.	12/02/2016	178.73	180.36	173.18	175.95	166.1	41775526
39.	15/02/2016	176.27	188.45	176.27	185	174.63	34130349
40.	16/02/2016	188.18	188.18	177.73	178.73	168.71	20895952
41.	17/02/2016	178.64	179.23	171.68	173.41	163.69	28792115
42.	18/02/2016	175.91	180.91	175.64	178.32	168.33	32260277
43.	19/02/2016	178	183.14	177.5	180.73	170.6	17617480
44.	22/02/2016	179.55	183	178.59	180.41	170.3	17567788

45.	23/02/2016	181.27	181.27	172.95	174.55	164.77	20001984
46.	24/02/2016	171.82	173.82	168.95	170	160.47	20654961
47.	25/02/2016	170.64	171.82	165.05	166.36	157.04	38340699
48.	26/02/2016	167.55	169.05	164.32	168	158.59	24687792
49.	29/02/2016	168	178.09	165.18	172.77	163.09	43510436
50.	01/03/2016	177.5	187.68	177.5	186.32	175.88	27223692

Table 39 : SBIN dataset

S.No	Date	Open	High	Low	Close	Adj Close	Volume
1.	14/05/2019	307.85	316.75	305.10	314.65	301.13	30521436.00
2.	15/05/2019	316.20	317.90	311.00	312.10	298.69	27932459.00
3.	16/05/2019	312.50	317.10	309.55	315.75	302.18	18147559.00
4.	17/05/2019	315.65	321.95	313.45	319.25	305.53	25838962.00
5.	20/05/2019	332.00	345.80	331.00	344.70	329.89	53207071.00
6.	21/05/2019	345.90	347.55	336.25	337.55	323.05	33457162.00
7.	22/05/2019	338.90	342.60	333.55	341.10	326.44	32002499.00
8.	23/05/2019	353.00	364.00	339.20	342.20	327.50	74413322.00
9.	24/05/2019	344.95	357.00	343.50	355.35	340.08	35703403.00
10.	27/05/2019	355.05	362.50	354.00	361.70	346.16	31374994.00
11.	28/05/2019	360.95	361.70	356.65	360.05	344.58	23501608.00
12.	29/05/2019	358.00	359.00	347.15	348.65	333.67	28370210.00
13.	30/05/2019	349.75	355.00	347.50	353.55	338.36	25712604.00
14.	31/05/2019	354.35	357.90	344.65	352.50	337.35	32588069.00
15.	03/06/2019	352.35	356.45	349.65	355.45	340.18	18380919.00
16.	04/06/2019	354.00	357.20	351.40	352.40	337.26	18130636.00
17.	06/06/2019	349.00	349.60	335.90	336.90	322.42	41941144.00
18.	07/06/2019	337.00	342.75	336.00	342.05	327.35	29739710.00
19.	10/06/2019	346.00	347.30	340.15	344.30	329.50	20500542.00
20.	11/06/2019	345.05	348.30	342.85	347.10	332.18	16281297.00

21.	12/06/2019	346.95	346.95	342.30	344.00	329.22	11564970.00
22.	13/06/2019	343.00	347.45	339.80	346.50	331.61	15509948.00
23.	14/06/2019	345.50	346.35	342.55	343.80	329.03	10347594.00
24.	17/06/2019	343.75	343.75	336.95	337.85	323.33	13779496.00
25.	18/06/2019	335.50	344.00	333.80	340.05	325.44	21957480.00
26.	19/06/2019	343.10	345.20	335.50	338.85	324.29	16416903.00
27.	20/06/2019	341.00	346.25	335.60	345.15	330.32	20626861.00
28.	21/06/2019	344.80	350.70	343.15	349.40	334.39	24749090.00
29.	24/06/2019	350.00	354.75	349.65	353.20	338.02	17919504.00
30.	25/06/2019	352.70	357.00	351.25	356.55	341.23	15922634.00
31.	26/06/2019	355.75	359.00	354.30	358.15	342.76	15835579.00
32.	27/06/2019	358.10	363.00	357.25	362.15	346.59	29658120.00
33.	28/06/2019	362.20	365.00	358.70	361.25	345.73	22708464.00
34.	01/07/2019	362.80	363.00	358.50	361.55	346.01	12348793.00
35.	02/07/2019	362.15	365.00	360.35	364.50	348.84	14482172.00
36.	03/07/2019	365.25	366.70	363.00	366.15	350.42	12626285.00
37.	04/07/2019	367.25	370.90	366.20	367.40	351.61	19252646.00
38.	05/07/2019	369.45	373.60	366.00	370.65	354.72	30019347.00
39.	08/07/2019	368.00	368.45	353.50	355.30	340.03	27247892.00
40.	09/07/2019	353.50	360.00	352.50	359.50	344.05	16579914.00
41.	10/07/2019	360.55	361.50	351.30	354.20	338.98	18681819.00
42.	11/07/2019	358.50	364.00	357.00	363.20	347.59	20048321.00
43.	12/07/2019	363.00	366.55	361.50	363.60	347.98	14661748.00

44.	15/07/2019	364.10	365.00	357.60	360.05	344.58	15665939.00
45.	16/07/2019	358.50	366.00	357.65	364.35	348.69	16251836.00
46.	17/07/2019	364.05	373.55	363.05	372.40	356.40	17545177.00
47.	18/07/2019	371.95	373.80	362.55	363.65	348.02	21691056.00
48.	19/07/2019	365.10	366.05	355.15	356.00	340.70	22213173.00
49.	22/07/2019	356.00	359.55	348.70	350.85	335.77	19682038.00
50.	23/07/2019	351.20	353.00	341.00	342.20	327.50	27956145.00

Table 40 : Nifty 50 dataset

S.No.	Date	Open	High	Low	Close	Adj Close	Volume
1.	01/04/2022	17436.9	17703.7	17422.7	17670.45	17670.45	291800
2.	04/04/2022	17809.1	18114.65	17791.4	18053.4	18053.4	345500
3.	05/04/2022	18080.6	18095.45	17921.55	17957.4	17957.4	283500
4.	06/04/2022	17842.75	17901	17779.85	17807.65	17807.65	328800
5.	07/04/2022	17723.3	17787.5	17623.7	17639.55	17639.55	308800
6.	08/04/2022	17698.15	17842.75	17600.55	17784.35	17784.35	274400
7.	11/04/2022	17740.9	17779.05	17650.95	17674.95	17674.95	251700
8.	12/04/2022	17584.85	17595.3	17442.35	17530.3	17530.3	266000
9.	13/04/2022	17599.9	17663.65	17457.4	17475.65	17475.65	245100
10.	18/04/2022	17183.45	17237.75	17067.85	17173.65	17173.65	376100
11.	19/04/2022	17258.95	17275.65	16824.7	16958.65	16958.65	401400
12.	20/04/2022	17045.25	17186.9	16978.95	17136.55	17136.55	286100
13.	21/04/2022	17234.6	17414.7	17215.5	17392.6	17392.6	285200
14.	22/04/2022	17242.75	17315.3	17149.2	17171.95	17171.95	262700
15.	25/04/2022	17009.05	17054.3	16888.7	16953.95	16953.95	275700
16.	26/04/2022	17121.3	17223.85	17064.45	17200.8	17200.8	261100
17.	27/04/2022	17073.35	17110.7	16958.45	17038.4	17038.4	265100
18.	28/04/2022	17189.5	17322.5	17071.05	17245.05	17245.05	312900
19.	29/04/2022	17329.25	17377.65	17053.25	17102.55	17102.55	336200
20.	02/05/2022	16924.45	17092.25	16917.25	17069.1	17069.1	278200

21.	04/05/2022	17096.6	17132.85	16623.95	16677.6	16677.6	310600
22.	05/05/2022	16854.75	16945.7	16651.85	16682.65	16682.65	265800
23.	06/05/2022	16415.55	16484.2	16340.9	16411.25	16411.25	300500
24.	09/05/2022	16227.7	16403.7	16142.1	16301.85	16301.85	288400
25.	10/05/2022	16248.9	16404.55	16197.3	16240.05	16240.05	283100
26.	11/05/2022	16270.05	16318.75	15992.6	16167.1	16167.1	284300
27.	12/05/2022	16021.1	16041.95	15735.75	15808	15808	314900
28.	13/05/2022	15977	16083.6	15740.85	15782.15	15782.15	369100
29.	16/05/2022	15845.1	15977.95	15739.65	15842.3	15842.3	217600
30.	17/05/2022	15912.6	16284.25	15900.8	16259.3	16259.3	295700
31.	18/05/2022	16318.15	16399.8	16211.2	16240.3	16240.3	290400
32.	19/05/2022	15917.4	15984.75	15775.2	15809.4	15809.4	313900
33.	20/05/2022	16043.8	16283.05	16003.85	16266.15	16266.15	252400
34.	23/05/2022	16290.95	16414.7	16185.75	16214.7	16214.7	293800
35.	24/05/2022	16225.55	16262.8	16078.6	16125.15	16125.15	249800
36.	25/05/2022	16196.35	16223.35	16006.95	16025.8	16025.8	243300
37.	26/05/2022	16105	16204.45	15903.7	16170.15	16170.15	314300
38.	27/05/2022	16296.6	16370.6	16221.95	16352.45	16352.45	274100
39.	30/05/2022	16527.9	16695.5	16506.15	16661.4	16661.4	251400
40.	31/05/2022	16578.45	16690.75	16521.9	16584.55	16584.55	651600
41.	01/06/2022	16594.4	16649.2	16438.85	16522.75	16522.75	249600
42.	02/06/2022	16481.65	16646.4	16443.05	16628	16628	236000
43.	03/06/2022	16761.65	16793.85	16567.9	16584.3	16584.3	245500

44.	06/06/2022	16530.7	16610.95	16444.55	16569.55	16569.55	233600
45.	07/06/2022	16469.6	16487.25	16347.1	16416.35	16416.35	233800
46.	08/06/2022	16474.95	16514.3	16293.35	16356.25	16356.25	243500
47.	09/06/2022	16263.85	16492.8	16243.85	16478.1	16478.1	205000
48.	10/06/2022	16283.95	16324.7	16172.6	16201.8	16201.8	189700
49.	13/06/2022	15877.55	15886.15	15684	15774.4	15774.4	225500
50.	14/06/2022	15674.25	15858	15659.45	15732.1	15732.1	225400

Table 41 : S & P BSE datasets

S.No	Date	Open	High	Low	Close	Adj Close	Volume
1.	5/31/2017	31222.51	31255.28	31107.48	31145.8	31145.8	13000
2.	06/01/2017	31117.09	31213.12	31062.02	31137.59	31137.59	8300
3.	06/02/2017	31205.37	31332.56	31190.4	31273.29	31273.29	8500
4.	06/05/2017	31274.74	31355.42	31198.22	31309.49	31309.49	9000
5.	06/06/2017	31420.85	31430.32	31172.55	31190.56	31190.56	13600
6.	06/07/2017	31252.71	31346.99	31172.98	31271.28	31271.28	9500
7.	06/08/2017	31316.91	31354.51	31193.77	31213.36	31213.36	14000
8.	06/09/2017	31196.86	31289.99	31087.28	31262.06	31262.06	9800
9.	06/12/2017	31225.43	31225.43	31044.28	31095.7	31095.7	6800
10.	6/13/2017	31091.1	31260.77	31062.34	31103.49	31103.49	13300
11.	6/14/2017	31147.69	31190.36	31054.94	31155.91	31155.91	12700
12.	6/15/2017	31222.89	31229.44	31026.48	31075.73	31075.73	12700
13.	6/16/2017	31160.47	31182.73	31017.18	31056.4	31056.4	13300
14.	6/19/2017	31168.98	31362.15	31163.35	31311.57	31311.57	7500
15.	6/20/2017	31392.53	31392.53	31261.49	31297.53	31297.53	8100
16.	6/21/2017	31302.18	31336.44	31193.61	31283.64	31283.64	35700
17.	6/22/2017	31351.53	31522.87	31255.63	31290.74	31290.74	10200
18.	6/23/2017	31352.57	31365.39	31110.39	31138.21	31138.21	89700
19.	6/27/2017	31194.68	31294.96	30847.08	30958.25	30958.25	15300
20.	6/28/2017	30988.87	31000.48	30798.7	30834.32	30834.32	9900

21.	6/29/2017	30910.97	31097.92	30794.61	30857.52	30857.52	12300
22.	6/30/2017	30824.97	30965.45	30680.66	30921.61	30921.61	9200
23.	07/03/2017	31156.04	31258.33	31017.11	31221.62	31221.62	8100
24.	07/04/2017	31331.21	31353.46	31166.37	31209.79	31209.79	8700
25.	07/05/2017	31272.72	31284.64	31177.78	31245.56	31245.56	6500
26.	07/06/2017	31298.42	31460.7	31264.86	31369.34	31369.34	10100
27.	07/07/2017	31373.52	31426.29	31286.62	31360.63	31360.63	6200
28.	07/10/2017	31510.62	31768.39	31471.41	31715.64	31715.64	37900
29.	07/11/2017	31789.5	31885.11	31718.48	31747.09	31747.09	9100
30.	07/12/2017	31813.24	31865.69	31731.43	31804.82	31804.82	8300
31.	7/13/2017	31896.23	32091.52	31892.63	32037.38	32037.38	10500
32.	7/14/2017	32099.93	32109.75	31897.87	32020.75	32020.75	7200
33.	7/17/2017	32053.98	32131.92	32037.21	32074.78	32074.78	10300
34.	7/18/2017	31775.54	31911.61	31626.44	31710.99	31710.99	21600
35.	7/19/2017	31882.8	31978.89	31793.72	31955.35	31955.35	11300
36.	7/20/2017	32033.82	32057.12	31859.5	31904.4	31904.4	11200
37.	7/21/2017	32035.88	32062.23	31808.93	32028.89	32028.89	18000
38.	7/24/2017	32100.22	32320.86	32058.33	32245.87	32245.87	11500
39.	7/25/2017	32350.71	32374.3	32196.86	32228.27	32228.27	14100
40.	7/26/2017	32255.99	32413.63	32226.08	32382.46	32382.46	12400
41.	7/27/2017	32519.44	32672.66	32325.33	32383.3	32383.3	10900
42.	7/28/2017	32381.36	32381.36	32104.66	32309.88	32309.88	11300
43.	7/31/2017	32412.2	32546.5	32324.45	32514.94	32514.94	13100

44.	08/01/2017	32579.8	32632.02	32462.25	32575.17	32575.17	9000
45.	08/02/2017	32641.58	32686.48	32394.89	32476.74	32476.74	10500
46.	08/03/2017	32502.55	32502.55	32194.58	32237.88	32237.88	12600
47.	08/04/2017	32191.12	32352.19	32107.99	32325.41	32325.41	11300
48.	08/07/2017	32377.8	32396.14	32235.82	32273.67	32273.67	10400
49.	08/08/2017	32341.05	32354.77	31915.2	32014.19	32014.19	9400
50.	08/09/2017	31926.14	31967.28	31731.91	31797.84	31797.84	9300

Table 42 : Microsoft dataset

S.No	Date	Open	High	Low	Close	Adj Close	Volume
1.	01/06/2023	325.93	333.53	324.72	332.58	330.5937	26773900
2.	02/06/2023	334.25	337.5	332.55	335.4	333.3969	25864000
3.	05/06/2023	335.22	338.56	334.66	335.94	333.9337	21307100
4.	06/06/2023	335.33	335.37	332.17	333.68	331.6872	20396200
5.	07/06/2023	331.65	334.49	322.5	323.38	321.4487	40717100
6.	08/06/2023	323.94	326.64	323.35	325.26	323.3175	23277700
7.	09/06/2023	324.99	329.99	324.41	326.79	324.8383	22514900
8.	12/06/2023	328.58	332.1	325.16	331.85	329.8681	24260300
9.	13/06/2023	334.47	336.98	330.39	334.29	332.2935	22951300
10.	14/06/2023	334.34	339.04	332.81	337.34	335.3253	26003800
11.	15/06/2023	337.48	349.84	337.2	348.1	346.0211	38899100
12.	16/06/2023	351.32	351.47	341.95	342.33	340.2855	46533600
13.	20/06/2023	339.31	342.08	335.86	338.05	336.0311	26375400
14.	21/06/2023	336.37	337.73	332.07	333.56	331.5679	25117800
15.	22/06/2023	334.12	340.12	333.34	339.71	337.6812	23556800
16.	23/06/2023	334.36	337.96	333.45	335.02	333.0192	23084700
17.	26/06/2023	333.72	336.11	328.49	328.6	326.6375	21520600
18.	27/06/2023	331.86	336.15	329.3	334.57	332.5719	24354100
19.	28/06/2023	334.66	337.98	333.81	335.85	333.8442	20259500
20.	29/06/2023	334.71	336.11	332.62	335.05	333.049	16997000

21.	30/06/2023	337.75	342.73	337.2	340.54	338.5062	26823800
22.	03/07/2023	339.19	340.9	336.57	337.99	335.9714	12508700
23.	05/07/2023	335.09	341.65	334.73	338.15	336.1305	18172400
24.	06/07/2023	337.3	342.99	335.5	341.27	339.2318	28161200
25.	07/07/2023	339.32	341.79	337	337.22	335.206	21185300
26.	10/07/2023	334.6	335.23	327.59	331.83	329.8482	32791400
27.	11/07/2023	331.06	332.86	327	332.47	330.4844	26698200
28.	12/07/2023	336.6	341.65	335.67	337.2	335.1862	29995300
29.	13/07/2023	339.56	343.74	339.02	342.66	340.6136	20567200
30.	14/07/2023	347.59	351.43	344.31	345.24	343.1781	28302200
31.	17/07/2023	345.68	346.99	342.2	345.73	343.6653	20363900
32.	18/07/2023	345.83	366.78	342.17	359.49	357.343	64872700
33.	19/07/2023	361.75	362.46	352.44	355.08	352.9594	39732900
34.	20/07/2023	353.57	357.97	345.37	346.87	344.7984	33778400
35.	21/07/2023	349.15	350.3	339.83	343.77	341.7169	69368900
36.	24/07/2023	345.85	346.92	342.31	345.11	343.0489	26678100
37.	25/07/2023	347.11	351.89	345.07	350.98	348.8839	41637700
38.	26/07/2023	341.44	344.67	333.11	337.77	335.7527	58383700
39.	27/07/2023	340.48	341.33	329.05	330.72	328.7449	39635300
40.	28/07/2023	333.67	340.01	333.17	338.37	336.3492	28484900
41.	31/07/2023	336.92	337.7	333.36	335.92	333.9138	25446000
42.	01/08/2023	335.19	338.54	333.7	336.34	334.3313	18311900
43.	02/08/2023	333.63	333.63	326.36	327.5	325.5441	27761300

44.	03/08/2023	326	329.88	325.95	326.66	324.7091	18253700
45.	04/08/2023	331.88	335.14	327.24	327.78	325.8224	23727700
46.	07/08/2023	328.37	331.11	327.52	330.11	328.1385	17741500
47.	08/08/2023	326.96	328.75	323	326.05	324.1028	22327600
48.	09/08/2023	326.47	327.11	321.05	322.23	320.3056	22373300
49.	10/08/2023	326.02	328.26	321.18	322.93	321.0014	20113700
50.	11/08/2023	320.26	322.41	319.21	321.01	319.0929	24342600

PUBLICATIONS

JOURNALS

- Published a paper titled"A Systematic Review on Artificial Neural Networks for Stock Market Prediction:in Tuijin jishu/Journal of Propulsion Technolgy(Scopus)ISSN 1001-4055 Vol 44No 6(2023)

- Published a Paper titled "Modified Extreme Learning Machine algorithm with deterministic Weight Modification for investment Decisions based on Sentiment Analysis" published in Recent advances in Computer Science and Communications indexed in Scopus ISSN(print):2666-2558(online)2666-2566 Vol 16,issue 8, 2023.
  DOI: 10.2174/2666255816666230815121119

- Published a paper titled "the Deterministic weight modification based Extreme Learning machine for stock price predictions " in Recent patents on Engineering indexed in Scopus ISSN(print)1872-2121(online)2212-4047 Vol 19,issue 2, 2023
  DOI: 10.2174/0118722121268858231111180830

CONFERENCE

- Presented and Published a paper entitled "Enhanced Extreme Learning Machine Algorithm with Deterministic Weight Modification for Investment Decision on Indian Stocks" in *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, 2022: IEEE, pp. 1409-1414.

- Presented and published a paper titled "Extreme Learning Machine Algorithm with Deterministic Weight Modification for investment Decision based on sentiment Analysis" in National Conference on Multidisciplinary for Sustainable Innovations (NCMRSI 2023) in Avinashilingam institute for Home Science ad Higher Education for Women on 10th Feburary 2023.

Avinashilingam Institute for HomeScience and Higher Education for Women
(Deemed to be University Estd. u/s 3 of UGC Act 1956, Category 'A' by MHRD
Re-accredited with A++ Grade by NAAC. CGPA 3.65/4, Category I by UGC
Coimbatore - 641 043, Tamil Nadu, India

Appendix L2

(Item No 5 of Check List) Details of Research Publications

S.No	Article	Journal	Other Details Vol/No/Page No/ Year	Published in UGC-CARE / Scopus Indexed/ Web of Science
1	Modified Extreme Learning Machine with Deterministic weight modification for investment decision based on sentiment Analysis	Recent Advances in computer science and communication	Vol 16, 8, 78-88 (11) 2023	Scoups indexed
2	Deterministic weight modification based Extreme Learning Machine for stock price prediction	Recent patents on Engineering	Vol 19, issue 2, Article in press DOI: 10.2174/011 872212126 88582311 1180830	Scopus indexed

*Proof of list of Journals from Internet to be attached along with copies of reprints.

Scholar

Supervisor

Checked By: 22/4/2024

HoD/Dean of Respective School

The scholar Miss. Kalaiselvi, K. has published her article in the following journals:

1. Recent advances in Computer Science and Communications is indexed and active in Scopus from 2020 to present

2. Recent Patents on Engineering indexed and active in Scopus from 2008 to present.

This may be considered.

J. [signature]
22.04.2024.

Recent Advances in Computer Science and Communications, 2023, 16, e150823219709

RESEARCH ARTICLE

BENTHAM SCIENCE

Modified Extreme Learning Machine Algorithm with Deterministic Weight Modification for Investment Decisions based on Sentiment Analysis

K. Kalaiselvi[1,*] and Vasantha Kalyani David[1]

[1]Department of Computer Science, Avinashilingam University for Women, Coimbatore, Tamilnadu, India

Abstract : *Background*: A significant problem in economics is stock market prediction. Due to the noise and volatility, however, timely prediction is typically regarded as one of the most difficult challenges. A sentiment-based stock price prediction that takes investors' emotional trends into account to overcome these difficulties is essential.

***Objective*:** This study aims to enhance the ELM's generalization performance and prediction accuracy.

***Methods*:** This article presents a new sentiment analysis based-stock prediction method using a modified extreme learning machine (ELM) with deterministic weight modification (DWM) called S-DELM. First, investor sentiment is used in stock prediction, which can considerably increase the model's predictive power. Hence, a convolutional neural network (CNN) is used to classify the user comments. Second, DWM is applied to optimize the weights and biases of ELM.

***Results*:** The results of the experiments demonstrate that the S-DELM may not only increase prediction accuracy but also shorten prediction time, and investors' emotional tendencies are proven to help them achieve the expected results.

***Conclusion*:** The performance of S-DELM is compared with different variants of ELM and some conventional method.

ARTICLE HISTORY

Received: May 18, 2023
Revised: June 13, 2023
Accepted: June 27, 2023

DOI:
10.2174/2666255816666230815121119

CrossMark

Keywords: Stock market prediction, Convolutional neural network, extreme learning machine, sentiment analysis, deterministic weight modification, machine learning.

1. INTRODUCTION

A significant amount of funds enters the stock market through the purchase of shares, improving the organic composition of commercial funds by encouraging resource attentiveness and significantly advancing the growth of the commodity economy. Stock prediction is typically the most complex challenge because of its characteristics of noise and volatility [1, 2]. In the social economy and organization of the modern world, the question of how to constantly predict the stock movement is quite unanswered. Numerous studies are considered subjects of economic science study in response to investor concerns and the attractiveness of big returns. In the earlier stage, the statistical approaches have been used to predict stock market data, such as autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), autoregressive conditional heteroscedasticity (ARCH), and generalized autoregressive conditional heteroskedasticity (GARCH) [3-6]. All of these models rely on the linearity of the prior and present variables. Since financial time series data typically lacks a defined pattern or linearity due to its chaotic and noisy character, statistical techniques typically struggle to effectively anticipate stock market indices. However, several negative stock market characteristics render predictions based on standard statistical approaches insufficient [7].

Machine learning (ML) models are now often used in the finance industry. Artificial neural networks (ANNs) [8, 9] achieve notable results and also well-known ML methods. The use of ANN in modeling economic situations is fast growing due to its intrinsic ability to detect complex nonlinear relationships in time series data based on historical data and correctly approximate any nonlinear function. ELM, a brand-new batch learning method, has just been suggested for training single hidden layer feed-forward neural networks (SLFNs'). The ELM algorithm primarily initializes hidden node values at random and calculates the output weights of SLFNs analytically. The key benefit of ELM is that SLFNs' hidden layer does not require tuning. ELM will result in the least squares solution of a system of line equations for the unknown output weights with the smallest norm property for the randomly selected input weights and hidden layer biases. With an incredibly quick learning rate, ELM has demon-

*Address correspondence to this author at the Department of Computer Science, Avinashilingam University for Women, Coimbatore, Tamilnadu, India; E-mail: selvi.vss@gmail.com

2666-2566/23 $65.00+.00

strated good generalization performances for many practical applications [10-13]. However, significant problems with the practical use of the ELM still exist, most notably how to select the optimal number of hidden nodes, typically accomplished through trial and error. More hidden neurons are typically needed for the ELM than for traditional tuning-based learning algorithms [14]. Hence, many research works are focused on finding optimal weights and biases for ELM to enhance the performance of prediction results [15, 16]. However, the conventional algorithms have many defects, such as a low convergence rate, long computational time, and local optima. Hence, our previous paper proposed that DWM was used to adjust the weight and bias of ELM, called DELM [17], which was applied to stock price prediction. The DELM produced higher accuracy when compared with some conventional and variant ELM.

On the other hand, the developed stock market prediction methods primarily use historical data as their input, ignoring other stock-impacting elements and their intricate talk into mechanisms. The stock market does not always follow systematic ideologies due to human inconsistency [18]. In the economic system, their emotional, psychological, and behavioral traits are crucial. Additionally, recent studies revealed that investor sentiment may play an important outcome in stock market investment. There is a significant indication that stockholders are not entirely rational, and shareholder relationships in the stock market are becoming easier and more frequent as social networks become further significant to people. As a result, an investor's attitude and decision-making processes may be affected by the sentiment and opinions of other investors and those stated on social media, which may also, to some extent, have an impact on the stock market [19]. By calculating binary sentiment indices for bullish and bearish attitudes, the proposed scheme takes investors' sentiment into account. In this paper, we present our CNN-based sentiment analysis algorithm for classifying stock market comments into bearish and bullish views. Then, enhanced ELM is used for predicting the stock price which is enhanced by DWM. The present research work aims to reduce prediction error, enhance accuracy, and make a fast-learning time. The presented research contributions are made as follows,

- The sentiment index (SI) is considered using a CNN-based classification model
- The enhanced ELM based on DWN is applied to predict the stock price
- Four stock market datasets are used to analyze the performance of S-DELM
- Four different performance analyzers are considered for analyzing the strength of the S-DELM method

Organizational paper is used for the following components: Section 2 defines the ELM, Section 3 converses the DELM, Section 4 converses CNN, Section 5 converses about S-DELM, Section 6 examines the experimental findings, and Section 7 analyses the research's conclusion.

2. METHODOLOGY

2.1. Related Works

The study of ML models for time series prediction has gained popularity in recent years. For example, Shobana *et al*. (2021) proposed new adaptive particle swarm optimization (APSO)-based LSTM. The suggested model uses LSTM to comprehend intricate patterns in textual input. By introducing the APSO, weights are improved, enhancing the LSTM's performance. The APSO helps the LSTM choose the best weight for the environment with less iteration by combining the opposition-based learning (OBL) approach and the PSO [20]. Londhe *et al*. (2022) suggested efficiently forecasting the sentiment using a hybridized version of the deep bidirectional LSTM (SoEo -BiLSTM). The BiLSTM classifier is used to execute both forward and backward directions of the hybridized bald eagle hunting strategy and coyote adaptation behavior [21]. Wu *et al*. (2022) [22] provided a stock price prediction technique called SI-LSTM and stockholder sentiment. Then, for non-traditional data, we employ the sentiment analysis approach based on CNN, which can determine the investors' SI. To forecast stock markets, Lin (2022) [23] combined structured data, such as trading data and technical indicators, with unstructured data from social media. The performance of forecasting models is heavily influenced by parameter selection. The LSTM employs the GA. Ahuja *et al*. (2021) [24] developed a new deterministic ELM for FFNN developed *via* multiple kernel learning by using (Gray level co-occurrence matrix) GLCM for feature extraction for improving ELM. With the objective kernel function being a linear combination of various base kernels, two formulations of the kernel ELM are introduced, such as deterministic multiple kernel learning and GLCM.

Wu *et al*. (2021) [22] sentiment indexed (SI) based-LSTM called SI-LSTM is a strategy for predicting stock prices that takes into account a variety of data sources and investor sentiment. The developed SI-LSTM for forecasting the China Shanghai A-share market incorporates sentiment index, technical indicators, and historical transaction data for stocks as the feature set. Tripathi *et al*. (2020) [25] suggested a novel activation function and an evolutionary strategy to obtain optimized weights and biases of ELM for predicting credit scores. Priya *et al*. (2018) [26] suggested a new hybrid method based on Weighted ELM and Weighted ELM with bacterial foraging optimization (BFO) for predicting Hepatitis disease. To handle classification data with an unbalanced nature of class distribution, weighted ELM is proposed. The basic goal of weighted ELM is to raise the classification rate by computing and assigning the relevant weight value to each training sample. To maximize the classification accuracy, the approach is additionally linked with the weighted ELM. The new stock price prediction approach developed by Li *et al*. [27] is based on a trading signal mining platform that employs ELM to combine two different types of market information sources, namely stock prices and market news.

Khuwaja *et al*. (2019) [28] developed a framework to convert attributes using phase space reconstruction (PSR) and extract the phase space correlations between them to model the price movement to anticipate the movement of the stock price. The suggested framework has been tested against 100 equities in the Borsa Istanbul (BIST) exchange. From the historical data used in the previous studies, twenty-five (25) technical indicators were taken into consideration as inputs for our approach. The PSR transformation is applied to the top two features from each category as chosen by the feature selection method. Das *et al*. (2021) [29] developed a new strategy for predicting stock market effectiveness; it was developed by combining a modified crow search algorithm (CSA) and ELM. By resolving 12 benchmark issues, it is demonstrated that the suggested improved CSA, known as Particle Swarm Optimization (PSO)-based Group orientated CSA (PGCSA), outperforms other current algorithms. The PGCSA method is employed to obtain pertinent weights and biases of ELM to increase the efficacy of traditional ELM. Using performance metrics, technical indicators, and the hypothesis test (paired *t*-test), it is possible to observe how the hybrid PGCSA ELM model affects the prediction of the closing price of seven different stock indices on the following day (Table **1**).

2.2. Extreme Learning Machine

ELM is an SLFN with a simple three-layer structure that consists of an input layer, an output layer, and a hidden layer with a significant number of nodes for nonlinear processing. Fig. (**1**) shows the framework of ELM. The hidden nodes are launched at random, network characteristics are fixed, and connection weights must be learned. When compared to FNN, the advantages of ELM have been proved on a wide range of challenges in diverse disciplines [30]. It is based on experimental risk minimization, where the biases and weights that are initially produced randomly characterize the weight at the output layer [31]. However, it provides more effective generalization at a faster learning rate. For N training samples (x_i, t_i), the data models $x_i \in R^n$ and target value $t_i \in R^m$. The mathematical formulation of ELM is as follows:

$$o_i = \sum_{i=1}^{L} w_i g_i (w_i . x_j + b_i), \ \ j = 1, 2, .., N \tag{1}$$

Where, - weights between hidden and input layer. β_i is the weight for layers of output and hidden. is the biased term of i^{th} the hidden node. activation functions. - is the output of ELM. L - number of hidden neurons. The N samples can be estimated with zero error. *i.e* $\sum_{i=1}^{N} \|z_i - y_i\| = 0$. If β_i, w_i and b_j exist such that,

$$Hw = o \tag{2}$$

Where,

$$H = \begin{pmatrix} g(w_1 . x_1 + b_1) & \mathrm{L} & g(w_H . x_1 + b_H) \\ : & & : \\ g(w_1 . x_N + b_1) & \dots & g(w_H . x_N + b_H) \end{pmatrix}_{N \times H} \tag{3}$$

The output layer is designed as below,

$O = \left[o_1, o_{2,\dots,} o_N\right]^T$ and $w = \left[w_1, w_{2,\dots,} w_L\right]^T$. - hidden layer output matrix. The existence of indicates that the ELM model with hidden nodes (L) can learn these training samples (N) so that,

$$t_j = \sum_{i=1}^{L} w_i g_i (w_i . x_j + b_i), \ \ j = 1, 2, .., N \tag{4}$$

Where, t_j is the target vector. The above equation can be written compactly in the matric vector form as follow,

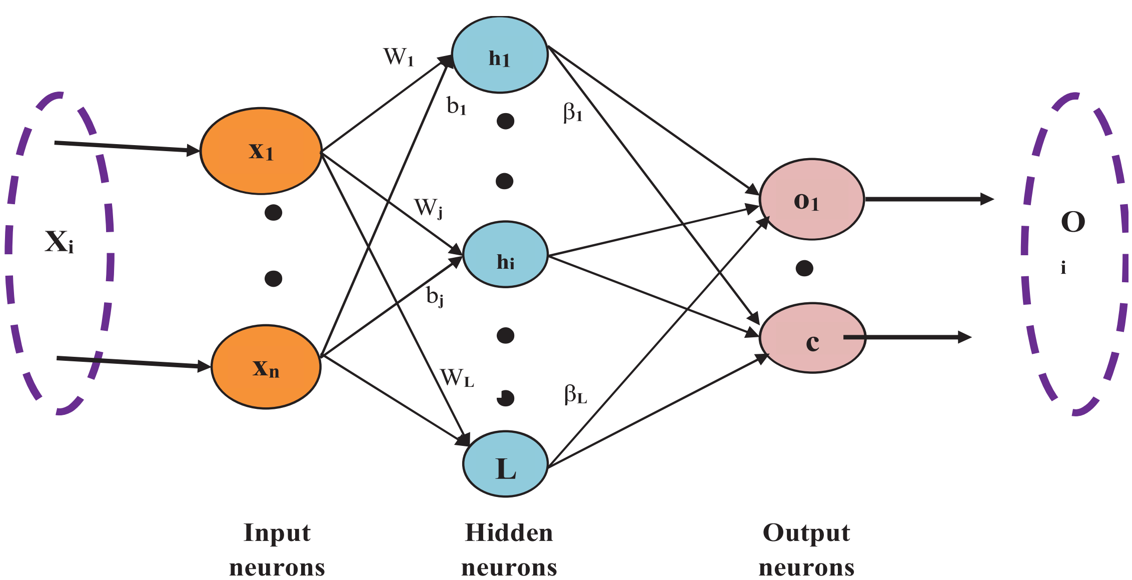

Fig. (1). Framework of ELM. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

Table 1. Technical Indicators.

S.No	Name of Technical Indicators	Formulas	Descriptions
1	Simple Moving Average (SMV)	$MV = \frac{x_1 + x_2 + \cdots + x_n}{n}$	The average value of specific $'t'$ days
2	10-days Moving Average	$MV_{10} = \frac{x_1 + x_2 + \cdots + x_n}{n}$	The average value of the last 10 transaction days
3	Momentum	$M = C_t - C_{t-4}$	It is measuring the sum of prices over given period length
4	Stochastic (K%)	$STCK = \frac{C_t - LL_{t-n}}{HH_{t-n} - LL_{t-n}} \times 100$	Stochastic determining the velocity of price variation. The comparative position of the current closing price in a certain period is calculated
5	Stochastic (D%)	$STCD = \frac{\sum_{i=0}^{n-1} K_{t-1\%}}{n}$	Specify the three days moving average
6	Relative Strength Index (RSI)	$= 100 - \frac{100}{1 + (\sum_{t=0}^{n-1} \mathrm{UP}_{t-1}/n)/(\sum_{t=0}^{n-1} \mathrm{DW}_{t=1}/n)}$	It measures the speed and movement of the price which ranges between 0 to 100.
7	Williams (%R)	$LW = \frac{H_n - C_t}{H_n - L_n} \times 100$	It computes overbought and oversold levels and is used to control market entry and exit opportunities.
8	Moving Average Convergence Divergence (MACD)	$= MACD(n)_{t-1} + \frac{2}{n+1} \; (Diff_t - MACD(n)_{t-1})$	MACD is to match up to the short-term and long-term momentum of a stock to calculate approximately its future direction
9	Commodity Channel Index (CCI)	$CCI = \frac{M_t - SM_t}{0.015 D_t} \times 100$	It assesses the present price level relative to an average price level over a certain length of time to determine a new movement or warn of severe conditions.
10	Price Oscillator (PO)	$\mathrm{PO} = \frac{\mathrm{MA}_5 - \mathrm{MA}_{10}}{\mathrm{MA}_5}$	PO is showing the relationship between two moving averages.

$$Hw = t \tag{5}$$

$t_j = [t_1, t_2, \ldots, t_N]$ is the target vector. The above equation becomes a linear parameter system because the input weights and hidden layer bias were determined randomly at the start of learning, and the least norm least squares solution of the linear constraint system is:

$$w = H^{\dagger} t \tag{6}$$

where $H^{\dagger}$ is the Moore–Penrose generalized inverse of H.

2.3. Deterministic-based Elm (DELM)

The ELM's weights and biases are altered using the DWM to boost generalization efficiency and prediction precision. During the learning process, DWM aims to controllably change the weights and bias. With the minimum error signal and fastest convergence time, it can find the global solution. By deterministically modifying the weights and bias to produce outputs that are nearly identical to preferred outputs, the aim is to reduce system error [32-34]. To get the outputs so close to the desired output, the network is given a weight modification. The predictable output for the next iteration is predetermined, and the learning procedure has retreated to ascertain the corresponding change in the weights. The solution is derived using certain approximations since the reverse computation takes too long. The output after this one will likely be close to what was anticipated after the change compared to the current iteration. Consider the current iteration to be the i^{th} iteration, and the output error for a definite output neuron, say $'O'$ is more than the MSE.

$$E_O(i) = \frac{1}{2}\sum_{p=1}^{P}[t_{po} - y_{po}(i)]^2 \geq \frac{E(i)}{O} \tag{7}$$

The aim of weights $w_{ko}(i+1)$ adjustment is to relate to this output $'O'$ close to the target output, which is defined as follows,

$$y_{PO}(i+1) = y_{PO}(i) + \beta\big(t_P(i) - y_{PO}(i)\big) = \beta t_{PO}(i) + (1-\beta) O_{PO} z \tag{8}$$

Where, $0 < \beta < 1$ for all data samples $'P'$.

The output is to obtain the following iteration's weights from the previous equations,

$$y_{pm}(i+1) = f\left(\sum_{k=1}^{K} w_{ko}(i+1)\,\bar{y}_{pk}(i+1)\right) \quad (9)$$

And thus,

$$\sum_{k=1}^{K} \Delta w_{ko}(i+1)\ \bar{y}_{pk}(i+1) = \varepsilon_{po} \quad (10)$$

Where,

$$\varepsilon_{po} = f^{-1}\left(y_{po}(i+1)\right) - \sum_{k=1}^{K} w_{ko}(i)\ \bar{y}_{pk}(i+1) \quad (11)$$

$$f^{-1}(x) = ln\left(\frac{x}{1-x}\right) \quad (12)$$

So, weight matrices to calculate $\Delta w_{ko}(i+1)$ as follows,

$$\Delta w_{ko}(i+1) = \frac{\sum_{P=1}^{P} e_{po}\,\bar{y}_{Pk}(i+1)}{\sum_{P=1}^{P} \bar{y}_{pk}^{2}(i+1)} \quad (13)$$

For each hidden neuron, both $\Delta w_{ko}(i+1)$ and errors are calculated as follows,

$$E_o^{(k)}(i+1) = \frac{1}{2}\sum_{p=1}^{P}\sum_{m=1}^{M}\left[t_{po} - y_{po}(i+1)\right]^2 \quad (14)$$

An error is choosing greater than the average error, which is then decreased thoroughly by affecting the results of the output layer. In the output neurons, t_{pm} is the target, and the expectable output is $y_{PO}(i) + \beta\left(t_P(i) - y_{PO}(i)\right)$. β is too near to one, which can cause unnecessary behavior. When β is set to 1, the target value will then be the predictable output. In addition, it is difficult for the system to produce the desired result in one cycle.

2.4. Convolutional Neural Network (CNN)

The sentiment analysis module in this study uses the CNN model presented by Kim to categorize comments [35]. The CNN is separated into three parts, such as input, convolution, and classification layer [36]. An is the text word vector matrix that works as the input layer, where is the number of distinctive phrases for each text and is a result of data processing. To convolve the word vector matrix, the convolutional layer initially passes over the convolution kernel W of length specifically:

$$t_i = f(w * s_{i,i+h-1} + b) \quad (15)$$

$s_{i,i+h-1}$ is a continuous text section made up of phrases i^{th} through phrase. * is the convolution operation. f and is the nonlinear function and bias term. Then, the training speed is normalized using the batch normalization (BN) algorithm. To lower the dimensionality and maintain a constant number of features, maximum value pooling is performed. The classification layer uses the BN algorithm to prevent changes in data distribution and the softmax layer to determine the classification probability. The probability is used to categorize stock market comments and is determined as follows:

$$P_j = P(y = j \mid X, b) = \frac{e^{X^T W_j + b_j}}{\sum_{i=1}^{L} e^{X^T W_j + b_j}} \quad (16)$$

P_j - denotes the probability of j^{th} class text. - is the input of the classification layers. W - is the weights matrix. and b_j is the offset element and bias term. L - is the number of classes.

3. PROPOSED SENTIMENT ANALYSIS BASED ON ENHANCED ELM

The proposed research work has two contributions, and Fig. (**2**) shows the flowchart of the proposed SDELM method. The benefits of the work are suggested, including investor sentiment in the model's stock prediction in order to increase its ability to predict. We specifically use sentiment analysis to determine the sentiment index from a huge number of stock market remarks that are categorized as either bullish or negative. The sentiment orientation of investors as a whole is reflected in these comments. After being inspired by this concept, we use a sentiment index, which accurately predicts shareholder behaviors, along with historical stock price data as the input for our stock market prediction engine. The stock closing prices, which can be impacted by many different factors and exhibit significant uncertainty and nonlinear properties, are predicted using the enhanced ELM. An initial goal is to perform a group that incorporates user sentiment opinions for the historical data about the stock as one element in predicting the price of the company. Based on the amount of daily bullish and bearish made by several users, the SI is created.

Consequently, to determine the group sentiment tendency and calculate the SI, we must first recognize the proper sentiment categorization of the individual stock review. Models called word2vec can un-supervise and learn semantic information from a lot of text. Words must be mapped from the old space to the new space for word2vec to perform. In particular, by learning the text, the word vector is created to represent the semantic information of each word, and the semantically related words are assigned to similar distances. In this study, softmax normalization and cosine similarity are observed using the Skip-gram in word2vec.

First, word2vec is developed to learn high-dimensional vector representations of phrases from large-scale stock comment corpora. The outcome is immediately applied if the phrase in the stock comments is to be categorized. If not, word2vec initializes it randomly. Following that, CNN will be given the word vectors, which denote the preprocessed text. Next, using a CNN enhanced by word2vec, we generate the SI for the group's sentiment analysis. Based on the total number of positive and negative comments made each day, the SI for the day is determined. It is shown by the group sentiment analysis, which might indicate the investor's general emotional inclination. To determine the daily SI for the stock, we employ the approach suggested by Antweiler and Frank [37].

$$BI_t = In\ 1{+}M_t^{bullish}\ /\ 1{+}M_t^{bearish} \quad (17)$$

where is the SI at the time t and and $1 + M_t^{bearish}$, which are determined by the number of bullish and bearish

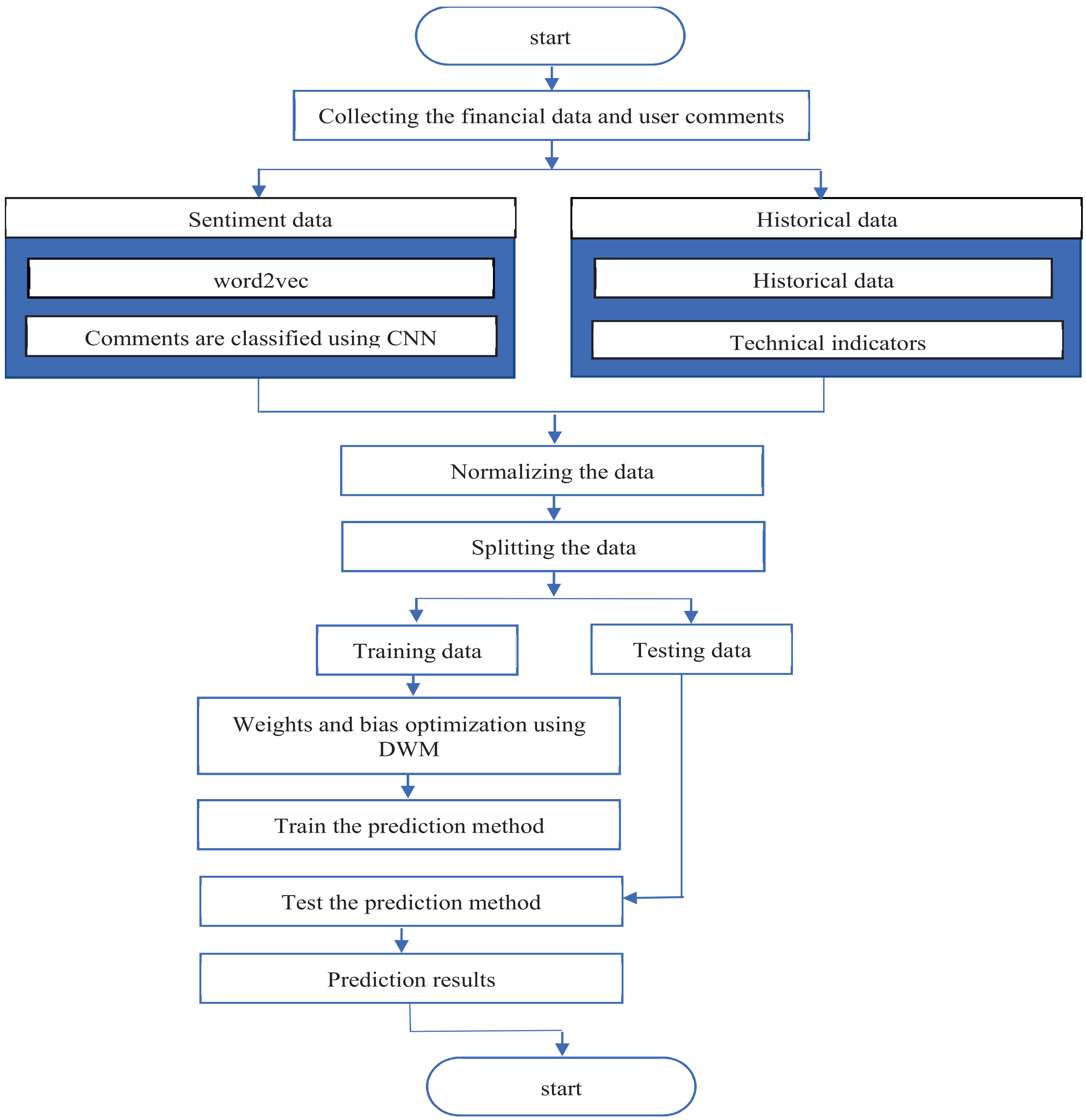

Fig. (2). Flowchart of proposed SDELM scheme. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

remarks, respectively. The index considers how bullish and bearish remarks have affected investor sentiment over time, and changes in the index's direction are proportionate to the weights given to bullish and bearish stock valuation views.

The index is positive, and the general tendency is characterized as bullish when more comments reflect bullishly. On the other hand, if more comments are expressing bearish sentiment than positive sentiment, the SI is negative and the overall mood is pessimistic. The size of the SI reveals the degree of inclination to a particular category and the positive or negative. reflects the group of sentimental orientation. Finally, the SI is considered as input to the prediction with their historical prices for predicting stock prices of the stock market using S-DELM.

4. EXPERIMENTAL RESULTS

This section tests the ability of the proposed sentiment-based DELM to estimate future prices using a set of standard stock datasets. It is believed that the value of the predictor with the lowest MAE and RMSE and high MAPE and R^2 has the best performance. The analysis findings are implemented using MATLAB R2015b. The performance of the S-DELM is related to some prediction methods, including DELM, GAELM, ELM, SVM, and BPNN over the four different datasets, namely S & P Sensex, Nifty 50, SBIN, and ICICI bank datasets. Four different performance measures are considered to analyse the ability of the S-DELM methods.

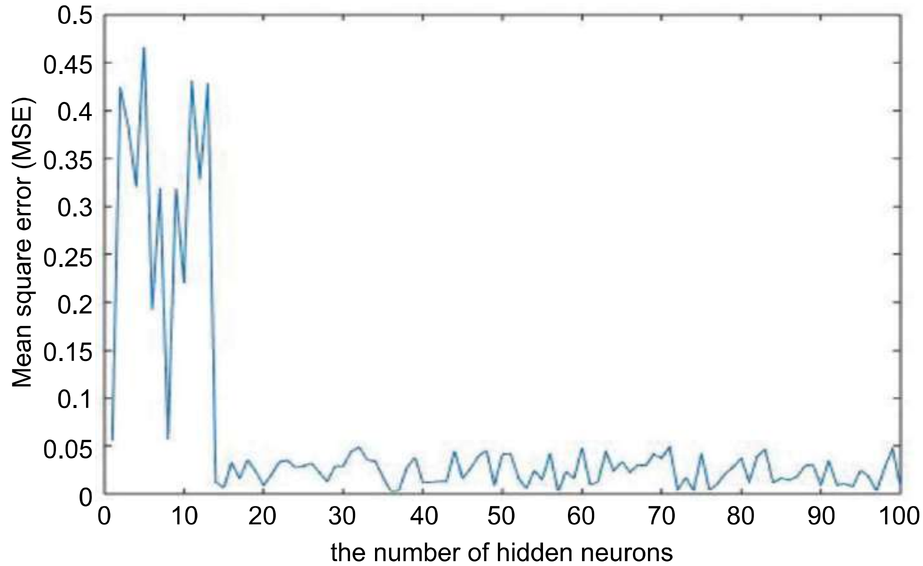

Fig. (3). Hidden neurons *vs.* MSE. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

4.1. Datasets

Two different experimental datasets are gathered, including the historical data and the comment dataset that calculates the SI. The stock comment dataset consists of a final comment that must be classified to determine the SI as well as comments for training the model. First, StockTwits "(https://stocktwits.com/)" comments made by stockholders were utilized to train the model. Investors have the option of marking the comment as bullish or bearish, even if StockTwits is a well-known public platform for depositors. As a result, it is possible to collect a great number of extremely precise comments.

For training purposes, we crawl the site's 96,903 comments. The historical datasets choose four features, including the open, high, low, and closing prices. Two stock indices are the S&P BSE Sensex and the Nifty 50. State Bank of India (SBIN) and ICICI Bank (ICICI) are two Indian stock bank prices that were retrieved from Yahoo [38]. Datasets between January'2015 and December'2022 are included. The gathered datasets are split into training and testing phases. About 30% of the total datasets are evaluated for testing, and the remaining 70% are considered for training. The scaling is done as follows

$$\bar{X}_i = \frac{X_i - X_{min}}{X_{max} - X_{min}}(\bar{X}_{max} - \bar{X}_{min}) + \bar{X}_{min} \tag{18}$$

Where, $\bar{X}_{min}$ - the minimum, $\bar{X}_{max}$ - maximum target value and are real input data.

4.2. Parameter Settings

A combination of layers, including input, hidden, and output layers, makes up the ELM algorithm. Based on specific applications, the input and output layer neurons are explained. The ten technical indicators discussed in this article can be considered input neurons for one output neuron, such as the closing price or index. The number of hidden layer neurons is important in the SLFN model, whereas the ELM method only has one hidden layer. The prediction effect will be reduced since few neurons will result in under-fitting, and too many will provide over-fitting. To determine a better number of nodes, this study continuously evaluates and updates the network's hidden layer node count. The ELM network's chosen maximum hidden neurons are 100. However, the optimal neurons are selected based on Fig. (**3**). According to Fig. (**3**), the learning model's MSE is the smallest when there are 35 hidden neurons. A stopping error of 0.0005 has been stated.

The results are computed from an average of 30 independent runs**.** The activation function of ELM also has another important role to decide prediction accuracy. The sigmoid function is employed as input and output phase activation functions. The maximum number of epochs is set to 1000. When choosing the best values, the two parameters in the DWM algorithm also determine the forecast accuracy. Hence, both values are set as based on [32].

4.3. Performance Measures

The performance is measured with four methods, namely Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-Square , which are determined as follows,

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{19}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|(y_i - \bar{y}_i)\right| \tag{20}$$

$$MAPE = \frac{100}{m}\sum_{i=1}^{m}\left|\frac{(y_i - \bar{y}_i)}{y_i}\right| \tag{21}$$

$$R^2 = 1 - \frac{\left(\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y}_i)^2\right)/N}{\left(\frac{1}{N}\sum_{i=1}^{N}(\bar{y}_i - \bar{y}_i)^2\right)/N} \tag{22}$$

Where, and are the target value and predicted output, respectively. is the total number of data points. y_i is the mean of real values. The value of R^2 is closer to 1, stronger ability and better the model.

4.4. Results Analysis

The stock market prediction is typically regarded as one of the most difficult issues in time-series prediction due to its characteristics of noise and volatility. The main goal of the current research work is sentiment-based DELM, which is covered in this section. The SI is calculated using a CNN-based classification algorithm which is used as an input with historical datasets for predicting the stock price using enhanced DELM, which is enhanced by DWM. The experimental results analysis is discussed in this present section. The results are shown in Tables **2-5**. The graphical representation is shown in Figs. (**4-7**) for S &P Sensex, Nifty 50, ICICI, and SBIN bank datasets, respectively. The results indicate that the S-DELM prediction method produced a high performance for all the datasets when compared with literature prediction methods. For all performance measurements, for instance, the S &P Sensex dataset is taken into account in this discussion part. The designed S-DELM technique produced a low RMSE score of 0.0095 when measured against all other algorithms. Moreover, 0.0109, 0.0122, 0.0134, 0.0149, and 0.0188 were similarly produced by the DELM, GAELM, ELM, SVM, and BPNN, respectively.

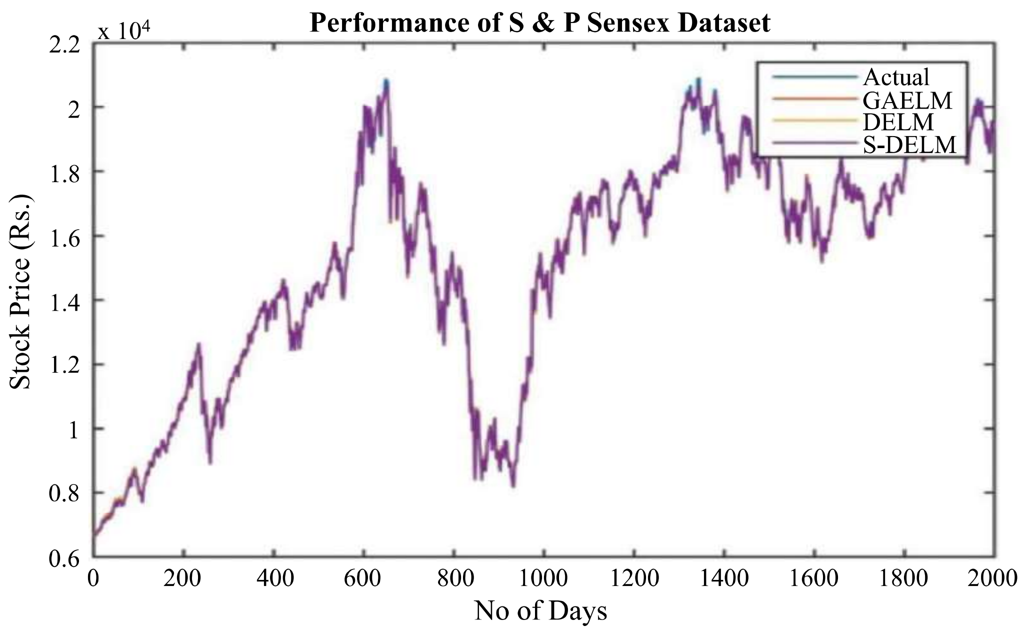

Fig. (4). Performance of the ICICI dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

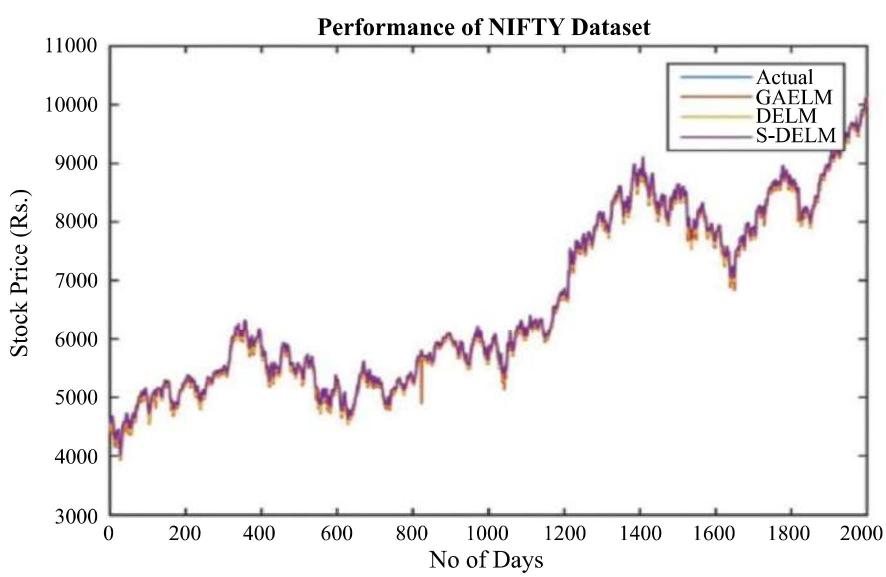

Fig. (5). Performance of the SBIN dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

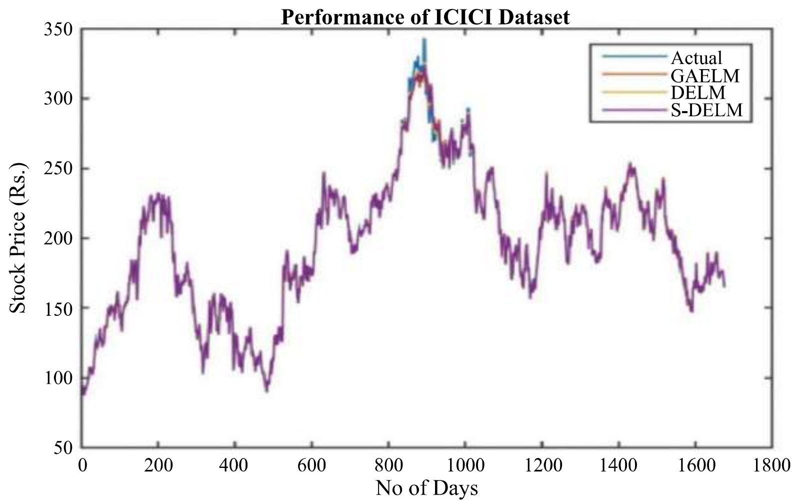

Fig. (6). Performance of the ICICI dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

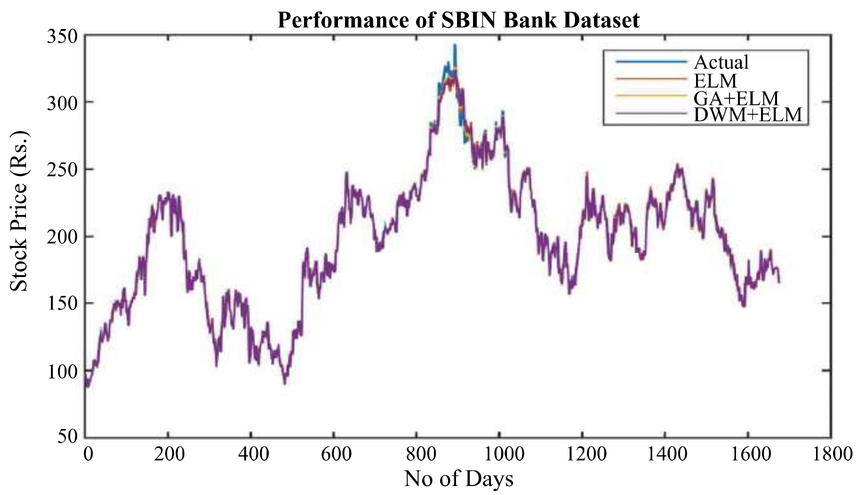

Fig. (7). Performance of the SBIN dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

The developed S-DELM technique produced a low MAE value of 0.0051 when measured against all other algorithms, 0.0056, 0.0064, 0.0073, 0.0089, and 0.0094 are the results of the DELM, GAELM, ELM, SVM, and BPNN, respectively. The designed S-DELM technique produced a high MAPE score of 7.67 when measured against all other algorithms. About 6.72, 5.08, 4.84, 5.08, and 5.67 were produced by the DELM, GAELM, ELM, SVM, and BPNN, respectively. The designed S-DELM technique produced a high MAPE score of 0.9794 when measured against all other algorithms. About 0.9537, 0.9381, 9072, 0.8567, and 0.7925 were produced by the DELM, GAELM, ELM, SVM, and BPNN, respectively.

According to the overall experimental results, the developed sentiment-based enhanced ELM called S-DELM produced high prediction accuracy and fast convergence. Human elements are not excluded as hypotheses in the proposed model; hence theoretical research includes behavioural analysis. According to our hypothesis, the stock price is significantly influenced by the investor's irrational behaviour in addition to the company's intrinsic worth. In other words, the price choice and its fluctuations in the securities market are significantly influenced by the psychology and behaviour of the investor. In our S-DELM model, sentiment analysis is the key to a bullish or bearish interpretation of stock reviews, and the addition of SI adds to the investor's general emotional inclination, considering variables that influence the stock price. The two components that contribute to the results' improvement are (1) behavioural finance, which considers the impact of emotional factors on stocks, and (2) better ELM with DWM, which pays greater consideration to obtain optimal weights and biases.

CONCLUSION

In this study, a sentiment-based DELM (S-DELM) for stock market forecasting is proposed. However, the ELM has many limitations due to random initialization of bias and weights, and human emotion regarding stock prediction can help enhance prediction accuracy. Hence, we combine user sentiment and DWN with ELM for predicting stock price

Table 2. Performance results of the Nifty 50 dataset.

Approaches	RMSE	MAE	MAPE	R^2
S-DELM	0.0095	0.0492	7.67	0.9794
DELM	0.0109	0.0563	6.72	0.9537
GAELM	0.0122	0.0649	5.08	0.9381
ELM	0.0134	0.0739	4.84	0.9072
SVM	0.0149	0.0089	5.08	0.8567
BPNN	0.0188	0.0094	5.67	0.7925

Table 3. Performance results of the S&P Sensex dataset.

Approaches	RMSE	MAE	MAPE	R^2
S-DELM	0.0493	0.3264	7.5417	0.9733
DELM	0.0585	0.0391	6.374	0.9672
GAELM	0.0758	0.0467	4.6781	0.9433
ELM	0.0836	0.0574	3.6756	0.8917
SVM	0.0873	0.0736	2.9547	0.8363
BPNN	0.0955	0.0844	2.4582	0.8224

Table 4. Performance results of the SBIN dataset.

Approaches	RMSE	MAE	MAPE	R^2
S-DELM	0.0094	0.0064	7.9517	0.9607
DELM	0.0201	0.0072	6.9475	0.9407
GAELM	0.0212	0.0078	6.2670	0.9164
ELM	0.0218	0.0084	5.7561	0.8568
SVM	0.0285	0.0078	4.0675	0.8105
BPNN	0.0389	0.0085	3.9472	0.7528

Table 5. Performance results of the ICICI dataset.

Approaches	RMSE	MAE	MAPE	R^2
S-DELM	0.0094	0.0049	5.2203	0.9561
DELM	0.0285	0.0058	4.8069	0.9217
GAELM	0.0257	0.0061	4.0881	0.8964
ELM	0.0385	0.0066	3.8347	0.8419
SVM	0.0495	0.0075	3.2619	0.8452
BPNN	0.0438	0.0088	2.8342	0.8104

prediction. Expressly, the SI is utilized to consider the investor's emotional tendencies. DWM is applied to optimize the connection weights of ELM. The experimental result clearly shows that the proposed S-DELM made high prediction accuracy when matched with literature algorithms.

LIST OF ABBREVIATION

ELM = Extreme Learning Machine

DWM = Deterministic Weight Modification

CNN = Convolutional Neural Network (CNN)

ML = Machine Learning

ANNs = Artificial Neural Networks

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data supporting the findings of the study are available within the article.

FUNDING

None.

CONFLICT OF INTEREST

The authors declares no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] K. Velusamy, and R. Amalraj, "Performance of the cascade correlation neural network for predicting the stock price", *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT),* 2017, pp. 1-6 Coimbatore, India http://dx.doi.org/10.1109/ICECCT.2017.8117824

[2] K. Kalaiselvi, K. Velusamy, and C. Gomathi, "Financial prediction using back propagation neural networks with opposition based learning", *J. Phys.: Conf. Ser.,* vol. 1142, no. 1, p. 012008, 2018. http://dx.doi.org/10.1088/1742-6596/1142/1/012008

[3] R. Efendi, N. Arbaiy, and M.M. Deris, "A new procedure in stock market forecasting based on fuzzy random auto-regression time series model", *Inf. Sci.,* vol. 441, pp. 113-132, 2018. http://dx.doi.org/10.1016/j.ins.2018.02.016

[4] L.Y. Wei, C.H. Cheng, and H.H. Wu, "A hybrid ANFIS based on n-period moving average model to forecast TAIEX stock", *Appl. Soft Comput.,* vol. 19, pp. 86-92, 2014. http://dx.doi.org/10.1016/j.asoc.2014.01.022

[5] Y.S. Lee, and L.I. Tong, "Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming", *Knowl. Base. Syst.,* vol. 24, no. 1, pp. 66-72, 2011. http://dx.doi.org/10.1016/j.knosys.2010.07.006

[6] C-F. Chen, W-H. Ho, H-Y. Chou, S-M. Yang, I-T. Chen, and H-Y. Shi, "Long-term prediction of emergency department revenue and visitor volume using autoregressive integrated moving average model", *Comput Math Methods Med,* vol. 2011, 2011. http://dx.doi.org/10.1155/2011/395690

[7] R. Dash, P.K. Dash, and R. Bisoi, "A self adaptive differential harmony search based optimized extreme learning machine for fi-

nancial time series prediction", *Swarm Evol. Comput.,* vol. 19, pp. 25-42, 2014.
http://dx.doi.org/10.1016/j.swevo.2014.07.003

[8] A.H. Moghaddam, M.H. Moghaddam, and M. Esfandyari, "Stock market index prediction using artificial neural network", *J. Econ. Finance Adm. Sci.,* vol. 21, no. 41, pp. 89-93, 2016.
http://dx.doi.org/10.1016/j.jefas.2016.07.002

[9] W. Huang, Y. Nakamori, and S.Y. Wang, "Forecasting stock market movement direction with support vector machine", *Comput. Oper. Res.,* vol. 32, no. 10, pp. 2513-2522, 2005.
http://dx.doi.org/10.1016/j.cor.2004.03.016

[10] J. Wang, S. Lu, S.H. Wang, and Y.D. Zhang, "A review on extreme learning machine", *Multimedia Tools Appl.,* vol. 81, no. 29, pp. 41611-41660, 2022.
http://dx.doi.org/10.1007/s11042-021-11007-7

[11] P. Vigneshvaran, and A.V. Kathiravan, "Heart disease prediction using an optimized extreme learning machine with bacterial colony optimization", *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC),* 2022 Trichy, India
http://dx.doi.org/10.1109/ICOSEC54921.2022.9952051

[12] X. Tang, and L. Chen, "Artificial bee colony optimization-based weighted extreme learning machine for imbalanced data learning", *Cluster Computing,* vol. 22, no. Suppl 3, pp. 6937-6952, 2019.
http://dx.doi.org/10.1007/s10586-018-1808-9

[13] J. Hu, Han, A.A. Heidari, Y. Shou, H. Ye, L. Wang, X. Huang, H. Chen, Y. Chen, and P. Wu, "Detection of COVID-19 severity using blood gas analysis parameters and Harris hawks optimized extreme learning machine", *Comput. Biol. Med.,* vol. 142, p. 105166, 2022.
http://dx.doi.org/10.1016/j.compbiomed.2021.105166 PMID: 35077935

[14] H. Chen, Q. Zhang, J. Luo, Y. Xu, and X. Zhang, "An enhanced bacterial foraging optimization and its application for training kernel extreme learning machine", *Appl. Soft Comput.,* vol. 86, p. 105884, 2020.
http://dx.doi.org/10.1016/j.asoc.2019.105884

[15] F. Weng, Y. Chen, Z. Wang, M. Hou, J. Luo, and Z. Tian, "Gold price forecasting research based on an improved online extreme learning machine algorithm", *J. Ambient Intell. Humaniz. Comput.,* vol. 11, no. 10, pp. 4101-4111, 2020.
http://dx.doi.org/10.1007/s12652-020-01682-z

[16] X. Xue, M. Yao, Z. Wu, and J. Yang, "Genetic ensemble of extreme learning machine", *Neurocomputing,* vol. 129, pp. 175-184, 2014.
http://dx.doi.org/10.1016/j.neucom.2013.09.042

[17] K. Kalaiselvi, and V.K. David, "Enhanced extreme learning machine algorithm with deterministic weight modification for investment decision on indian stocks", *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC),* 2022pp. 1409-1414 Trichy, India
http://dx.doi.org/10.1109/ICOSEC54921.2022.9951899

[18] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM", *Neural Comput. Appl.,* vol. 32, no. 13, pp. 9713-9729, 2020.
http://dx.doi.org/10.1007/s00521-019-04504-2

[19] F. Wang, Y. Zhang, Q. Rao, K. Li, and H. Zhang, "Exploring mutual information-based sentimental analysis with kernel-based extreme learning machine for stock prediction", *Soft Comput.,* vol. 21, no. 12, pp. 3193-3205, 2017.
http://dx.doi.org/10.1007/s00500-015-2003-z

[20] J. Shobana, and M. Murali, "Adaptive particle swarm optimization algorithm based long short-term memory networks for sentiment analysis", *J. Intell. Fuzzy Syst.,* vol. 40, no. 6, pp. 10703-10719, 2021.
http://dx.doi.org/10.3233/JIFS-201644

[21] D. Londhe, and A. Kumari, "Multilingual sentiment analysis using the social eagle-based bidirectional long short-term memory", *Int. J. Intell,* vol. 15, no. 2, pp. 479-493, 2022.

[22] S. Wu, Y. Liu, Z. Zou, and T.H. Weng, "SI-LSTM: Stock price prediction based on multiple data sources and sentiment analysis", *Connect. Sci.,* vol. 34, no. 1, pp. 44-62, 2022.
http://dx.doi.org/10.1080/09540091.2021.1940101

[23] Y.L. Lin, C.J. Lai, and P.F. Pai, "Using deep learning techniques in forecasting stock markets by hybrid data with multilingual sentiment analysis", *Electronics,* vol. 11, no. 21, p. 3513, 2022.
http://dx.doi.org/10.3390/electronics11213513

[24] B. Ahuja, and V.P. Vishwakarma, "Deterministic Multi-kernel based extreme learning machine for pattern classification", *Expert Syst. Appl.,* vol. 183, p. 115308, 2021.
http://dx.doi.org/10.1016/j.eswa.2021.115308

[25] D. Tripathi, D.R. Edla, V. Kuppili, and A. Bablani, "Evolutionary extreme learning machine with novel activation function for credit scoring", *Eng. Appl. Artif. Intell.,* vol. 96, p. 103980, 2020.
http://dx.doi.org/10.1016/j.engappai.2020.103980

[26] S. Priya, and R. Manavalan, "Optimum parameters selection using bacterial foraging optimization for weighted extreme learning machine", *J. Soft Comput.,* vol. 8, no. 4, 2018.

[27] X. Li, H. Xie, R. Wang, Y. Cai, J. Cao, F. Wang, H. Min, and X. Deng, "Empirical analysis: Stock market prediction *via* extreme learning machine", *Neural Comput. Appl.,* vol. 27, no. 1, pp. 67-78, 2016.
http://dx.doi.org/10.1007/s00521-014-1550-z

[28] P. Khuwaja, S.A. Khowaja, I. Khoso, and I.A. Lashari, "Prediction of stock movement using phase space reconstruction and extreme learning machines", *J. Exp. Theor. Artif. Intell.,* vol. 32, no. 1, pp. 59-79, 2020.
http://dx.doi.org/10.1080/0952813X.2019.1620870

[29] S. Das, T.P. Sahu, R.R. Janghel, and B.K. Sahu, "Effective forecasting of stock market price by using extreme learning machine optimized by PSO-based group oriented crow search algorithm", *Neural Comput. Appl.,* vol. 34, no. 1, pp. 555-591, 2022.
http://dx.doi.org/10.1007/s00521-021-06403-x PMID: 34413575

[30] F. Zhang, "Extreme learning machine for stock price prediction", *Int. J. Electr. Eng. Educ.,* 2021.
http://dx.doi.org/10.1177/0020720920984675

[31] G.B. Huang, Q.Y. Zhu, and C.K. Siew, "Extreme learning machine: Theory and applications", *Neurocomputing,* vol. 70, no. 1-3, pp. 489-501, 2006.
http://dx.doi.org/10.1016/j.neucom.2005.12.126

[32] S.C. Ng, C.C. Cheung, and S.H. Leung, "Magnified gradient function with deterministic weight modification in adaptive learning", *IEEE Trans. Neural Netw.,* vol. 15, no. 6, pp. 1411-1423, 2004.
http://dx.doi.org/10.1109/TNN.2004.836237 PMID: 15565769

[33] C. Wang, and D.J. Hill, *Deterministic learning theory for identification, recognition, and control..* CRC Press, 2009.

[34] K. Velusamy, and R. Amalraj, "Cascade correlation neural network with deterministic weight modification for predicting stock market price", *IOP Conference Series: Materials Science and Engineering,* vol. vol. 1110, 2021p. 012005
http://dx.doi.org/10.1088/1757-899X/1110/1/012005

[35] K. O'Shea, and R. Nash, "An introduction to convolutional neural networks", *arXiv:1511.08458,* 2015.

[36] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks", *Pattern Recognit.,* vol. 77, pp. 354-377, 2018.
http://dx.doi.org/10.1016/j.patcog.2017.10.013

[37] W. Antweiler, and M.Z. Frank, "Is all that talk just noise? The information content of internet stock message boards", *J. Finance,* vol. 59, no. 3, pp. 1259-1294, 2004.
http://dx.doi.org/10.1111/j.1540-6261.2004.00662.x

[38] "Yahoo finance", https://finance.yahoo.com/

Send Orders for Reprints to reprints@benthamscience.net

Recent Patents on Engineering, 2023,19,*e041223224185*

RESEARCH ARTICLE

Deterministic Weight Modification-based Extreme Learning Machine for Stock Price Prediction

K. Kalaiselvi[1,*] and Vasantha Kalyani David[1]

[1]*Department of Computer Science, Avinashilingam University for Women, Coimbatore, Tamilnadu, India*

ARTICLE HISTORY

Received: August 13, 2023
Revised: September 06, 2023
Accepted: September 08, 2023

DOI:
10.2174/0118722121268858231111180830

Abstract: ***Background:*** The prediction of the stock price is considered to be one of the most fascinating and important research and patent topics in the financial sector.

Aims: Making more accurate predictions is a difficult and significant task because the financial industry supports investors and the national economy.

Objectives: The DWM is used to adjust the connection weights and biases to enhance prediction precision and convergence rate. DWM was proposed as a method to reduce system error by changing the weights of various levels. The methods for predictable changes in weight were provided together with the computational difficulty.

Methods: An extreme learning machine (ELM) is a fast-learning method for training a single-hidden layer neural network (SLFN). However, the model's learning process is ineffective or incomplete due to the randomly chosen weights and biases of the input's hidden layers. Hence, this article presents a deterministic weight modification (DWM) based ELM called DWM-ELM for predicting the stock price.

Results: The calculated results showed that DWM-ELM had the best predictive performance, with RMSE (root mean square error) of 0.0096, MAE (mean absolute error) of 0.0563, 0.0428, MAPE (mean absolute percentage error) of 1.7045, and DS (Directional Symmetry) of 89.34.

Conclusion: The experimental results showed that, in comparison to other well-known prediction algorithms, the suggested DWM+ELM prediction model offers better prediction performance.

Keywords: Stock price prediction, prediction accuracy, convergence rate, extreme learning machine, deterministic weight modification, mean absolute error.

1. INTRODUCTION

The topic of stock price forecasting frequently arises in finance, statistics, computer science, and commerce. The stock market analysis, which involves compiling, organizing, and integrating a variety of important data, can help in comprehending and predicting the direction of stock prices as well as in selecting the best investments to minimize risks and maximize benefits. We can manage the stock market effectively and direct it in a healthy direction, which will be a powerful support for the economy's long-term growth if we can accurately and quickly forecast peak movements and stock prices [1]. As a result, studying stock forecasts can assist investors in choosing profitable ventures that enhance both the nation's and the world's economies. Forecasting stock price is challenging due to the stock's stochastic nature and unpredictable behavior, and time series analysis of data appears to be ineffective in capturing discontinuities, nonlinearity, and large dataset complexity [2].

Early stock price predictions are made using statistical approaches. So, it is impossible to identify the complexity of stock market businesses using typical statistical methods [3, 4]. Stock analysts and investors have looked at several machine learning (ML) methods for stock price prediction and making different trading decisions to address the limitations of the aforementioned statistical methodologies. Many ML techniques are used to replace traditional statistical techniques [5-7]. The artificial neural network (ANN) is a category of ML and has been applied for stock prediction [8]. ANN is the most proficient technique and has been found to be more effective in handling nonlinearity, and many methods are applied to forecast stock prices [9-14]. However, due to the complexity of the stock market data, the old method has some drawbacks, including a low convergence rate, poor prediction accuracy, and potential for falling into local optima. The weights and biases, which are based on the integration of random weights and biases, also affect how well the

*Address correspondence to this author at the Department of Computer Science, Avinashilingam University for Women, Coimbatore, Tamilnadu, India; E-mail: selvi.vss@gmail.com

prediction model performs. Hence, for an effective result, it has to be tuned better.

Huang *et al*. proposed a new technique to train SLFNs, and the resulting model was called the ELM. The ELM has many advantages, such as high training speed, simple structure, and high generalization ability [15]. It also prevents a lot of problems compared to the other gradient-based learning methods. In this model, hidden biases and connection weights between input and hidden layers are both initialized with random numbers. To determine the connection weights between hidden and output layers, the Moore-Penrose inverse is utilized (referred to as hidden-output link weights). Hence, it does not require iterative training and may be trained in a single step. Due to the weights and biases' random initialization, it occasionally may not deliver the best training accuracy, leading to poor testing accuracy. Moreover, the hidden layer needs more neurons than other SLFNs, which lengthens the testing period for untested samples.

In this study, the DWM algorithm is designed to improve ELM performance by modifying weights and biases for quick learning and strong global search capabilities. Throughout the ELM training process, it is intended to adjust the network weights and bias in a deterministic manner. By deterministically changing the weights of a multilayered feedforward neural network, ELM with DWM is utilized to decrease system error. The worst input-output pair, or the input pattern and its associated output that produces the biggest mean squared error, is chosen, and the weights of that pair are adjusted by taking into account the predicted output and system error. According to simulation data, DWM+ELM typically performs better than other popular modified ELM approaches in terms of convergence rate and percentage of global convergence. Finding the optimum combination of weights and biases would help the ELM model perform better, which is the objective of optimizing weights and biases in ELM for stock market prediction. The main goal is to reduce the loss function or prediction error. This entails identifying the optimal weights and biases that produce the most precise stock price forecasts or related financial metric projections. The research's contributions include the following:

- The DWM is utilized to improve ELM abilities and prediction accuracy.
- DWM+ELM will eliminate the randomness of the original ELM by evaluating the input weights and biases rather than assigning them at random.
- Applying the DWM+ELM strength analysis to four distinct stock market datasets.
- The performance of the prediction algorithm is measured using different performance metrics and compared with benchmark prediction methods.

The paper is systematized as follows: Section 2 of the study discusses relevant works. In section 3, the architecture of ELM is defended. In section 4, the DWM is described in great detail. Section 5 of the recommended DWM+ELM is covered. The experimental results are described in section 6, and the conclusion is presented in section 7.

2. RELATED WORKS

Stock prediction is a most motivating research area since successful predictions of the market's future program yield significant gains. ML algorithms have been used in several scientific projects, including stock market prediction. For instance, Weng *et al*. (2020) [16] developed an improved online ELM used to forecast gold prices. The global optimization of GA can successfully determine the output layer weight and hidden layer threshold. The created optimized ELM has a high level of performance, as seen by its low RMSE of 7.5, high MAPE of 0.0043, and high MdAE of 4.0925.

Li *et al*. (2021) [17] presented an enhanced CS approach to improve the efficacy of employing renewable energy, which is recommended for accurately forecasting short-term wind output. The adaptive flying step adjustment method enhances the capacity for both regional development and global exploration. The periodic flight migration technique also keeps the population variation of the CSA algorithm from being stranded in regional extremes. The suggested model's RMSE and MAPE values were kept below 20% and 4%, respectively.

Das *et al*. (2021) [18] developed a stock market prediction approach by incorporating a modified CSA and ELM. The PSO-based Group-oriented CSA (PGCSA) method, a suggested modification of the CSA, was used to obtain appropriate weights and biases of ELM to increase the efficacy of ELM. Using performance metrics, technical indicators, and hypothesis testing, the effect of the hybrid PGCSA-ELM on the prediction of the closing price of seven distinct stock indices on the following day was observed (paired t-test). The developed method achieved high performance, such as MAE of 16.39, MSE of 410.53, and MAPE of 0.9264.

Tripathi *et al*. [19] (2020) developed a new mixed-kernel-based ELM called MKELM for binary classification. First, a linear combination of the RBF and the polynomial kernel was employed. Then, an MKELM-RF (MKELM-based Random Forest Binary Classifier) was built. The bat optimization technique was developed to find a unique activation function with an evolutionary method to obtain optimum weights and biases. The developed technique achieved 86.98% accuracy, 86.94 sensitivity, 87.25 specificity, and 86.95 g-mean.

Chandar *et al*. (2020) [20] developed a new method of predicting stock market prices based on an optimized Elman neural network (ENN) using Grey Wolf optimization (GWO), which is used to determine the ENN parameters. Optimized ENN was utilized to predict the future price of stock data 1 day in advance, which obtained the very lowest MSE, such as 0.0009.

Selvamuthu *et al*. (2019) [21] developed a new stock market prediction model based on ANN-optimized different learning methods. The performance of the Levenberg-Marquardt (LM), Scaled Conjugate Gradient (SCG), and Bayesian Regularization (BR)-based neural networks for stock market prediction using tick data and 15-minute data of an Indian business was compared. In comparison to findings produced with tick data, the accuracy across a 15-minute dataset lowered to 96.2%, 97.0%, and 98.9% for LM, SCG, and BR, respectively.

Wu *et al*. (2021) proposed a combination of ELM- and DWT (discrete wavelet transformation)-based denoising to predict the trend [22]. DWT was used to remove noise from stock market data then denoised data was considered as inputs to the ELM. The forecast results of the suggested technique provided a high accuracy of 96.55%, which is a solid demonstration of the effectiveness of DWT-based denoising for stock movements.

Tang *et al*. (2020) [23] developed a new optimization ELM based on the differential evolution (DE) method for stock index prediction. The designed hybrid model applied ensemble empirical mode decomposition (EEMD) technology to the residual item after applying variational mode decomposition (VMD) to the stock index price to produce different modal components. The final prediction results were then obtained by superimposing the DE-ELM model's predictions for each modal factor and the residual element. The developed method produced high performance based on different performance measures with eRMSE as 22.6078, eMAE as 16.8280, and eMAPE as 0.0057.

Khuwaja *et al*. [24] (2019) developed a new stock movement prediction method based on ELM and PSR (Phase space reconstruction). The feature transformation method used by the framework, which calculates the information distance from the changed features, reflects the novelty of the framework. To forecast the movement of the stock price, the distance from the phase space dimensions was modeled using ELM. According to the experimental findings, the framework increased F-measure values for predictive performance by 4.5%.

Cui *et al*. (2022) [25] developed a new forecasting short-term traffic flow method based on a two-stage hybrid ELM model. To increase the effectiveness of the global optimum value search, the initial population distribution of the gravitational search algorithm (GSA) was determined using the PSO. The hybrid forecasting model was trained in a data-driven manner in the second stage using the outcomes of the first stage rather than the network structure parameters that the ELM produced at random. The suggested model's respective RMSEs were 288.03, 204.09, 220.52, and 163.92, while its respective MAPEs were 11.53%, 10.16%, 11.67%, and 12.02%. Experimental findings showed that our suggested model performed better than others.

Chen *et al*. (2022) [26] used the chaotic and complicated carbon futures price prediction to illustrate the effectiveness of this three-stage forecasting technique. In order to break down the price of carbon futures into a few intrinsic mode functions (IMFs) and one residue, we first used the EEMD approach. The IMFs and residue were then rebuilt using the fuzzy entropy and K-means clustering techniques, yielding three reconstructed components: a high-frequency series, a low-frequency series, and a trend series. Third, the stationary high and low-frequency series were implemented using the ARMA model, whilst the stationary trend series were done using the ELM model. According to empirical findings, the suggested reconstruction algorithm can increase prediction accuracy by more than 40% when compared to the conventional fine-to-coarse reconstruction method within the same forecasting framework.

The performance prediction accuracy, low convergence rate, and potential for both overfitting and underfitting are all greatly impacted by the ELM parameters. Hence, tuning the ELM parameters is, therefore, a crucial step in achieving improved generalisation performance. The literary approach performs quite well, but it suffers from a number of shortcomings, such as poor convergence rates, local optima, and premature convergence.

3. EXTREME LEARNING MACHINE

ELM is an effective SLFN and has the ability to handle non-linear information. In ELM, hidden nodes are deployed at random, network characteristics are fixed, and connection weights must be learned. The advantages of ELM have been proved on a wide range of challenges in various domains. It is based on experimental risk minimization, where the biases and weights that are initially produced randomly characterize the connection weight at the output layer of SLFNs [27]. So far, it provides more effective generalization at a faster learning rate. An input layer serves as the foundation of ELM and accepts the features or data points as input. A hidden layer with a set number of neurons or units (commonly referred to as hidden nodes) is introduced by ELM. The user selected this quantity as a hyperparameter. The buried layer neurons' weights and biases are created at random. These weights are normally chosen at random, either uniformly or according to a Gaussian distribution. By applying the activation function to a linear combination of its input characteristics and associated weights, each hidden neuron calculates its output. The final predictions made by an ELM generally come from a single output layer, which weights the outputs of the hidden layer neurons.

As ELM does not need iterative optimization like gradient descent and as the hidden layer weights are initialized at random, it is especially appealing. This makes it simple to implement and computationally effective. However, depending on the activation function used and the quantity of hidden neurons, its performance can change and needs to be customized for certain issues. The data points are denoted as $x_i \in R^n$, and the target value is represented as $t_i \in R^m$. The hidden nodes $(\widetilde{N})$ and activation function $g(x)$, which are arithmetically considered as follows:

$$\sum_{i=1}^{H} \beta_i \, g_i(w_i . x_j + b) = z_i (i = 1,2, \ldots . . m), j = 1,2, \ldots . n \quad (1)$$

Where, the connection weights between the hidden and input layer are represented by w_i. β_i- is the weight for layers of output and hidden. b - is the hidden bias. The N samples can be similar to SLFN with zero error, *i.e.*, $\sum_{i=1}^{N} \| z_i - y_i \| = 0$. If β_i, w_i and b_j exist such that:

$$\sum_{i=1}^{H} \beta_i \, g_i(w_i . x_j + b) = o_j \; (i = 1,2,3,4, \ldots . . , n) \quad (2)$$

The formula is reformed as:

$$H\beta = T \quad (3)$$

The output matrix (H) is denoted as:

$$H = \begin{bmatrix} g(w_1, x_1 + \boldsymbol{b}_1) & \ldots\ldots\ldots & g(w_{\tilde{N}}, x_1 + \boldsymbol{b}_{\tilde{N}}) \\ \vdots & \vdots & \vdots \\ g(w_1, x_{\tilde{N}} + \boldsymbol{b}_1) & \ldots\ldots\ldots & g(w_{\tilde{N}}, x_N + \boldsymbol{b}_{\tilde{N}}) \end{bmatrix}_{N\times m} \quad (4)$$

$$H = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_N^T \end{bmatrix}_{\tilde{N}\times m} \quad (5)$$

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times m} \quad (6)$$

The $\hat{w}_i$, $\hat{b}_i$ and $\hat{\beta}_i$ should be employed to train the SLFN such that:

$$\left\|H(\hat{w}_i, \hat{b}_i).\hat{\beta} - T\right\| = \min_{w,b,\beta}\|H(w_i, b_i).\beta - T\|(i = 1,2,..N) \quad (7)$$

The cost minimization function is as follows:

$$E = \sum_{j=1}^{N}\left(\sum_{i=1}^{\hat{N}}\beta_i g_i(w_i x_j + b_i) - t_j\right)^2 \quad (8)$$

Nonetheless, the ELM is a rather quick learning method; thus, iterative parameter tuning is not necessary. Once the input weights and hidden bias have been allocated at random, the hidden output matrix H is unaffected. The SLFN can be turned into a linear system solution ($H\beta = T$). The output matrix weight β can be defined as follows:

$$\hat{\beta} = H \dagger T \quad (9)$$

Where, $H \dagger$ is the Moore - Penrose generalized inverse of the matrix (H) [27].

4. DWM ALGORITHM

The ELM's weights and biases are modified using the DWM to enhance generalization efficiency and prediction precision. During the learning process, DWM aims to adaptively change the network weights and biases. With the minimum error signal and fastest convergence time, it can find the global solution. By deterministically modifying the weights and bias to produce outputs that are nearly identical to desired outputs, the aim is to reduce system error [28-30]. To get the outputs so close to the desired output, the network is given a weight modification. The predicted output for the next iteration is predetermined, and the learning procedure is reversed to ascertain the corresponding change in the weights. The solution is derived using certain approximations since the reverse computation takes too long. It is predicted that the output in the subsequent iteration will be close to the expected output following the modification compared to the existing iteration. If the present iteration is studied to be the ith iteration and the output error for a certain output neuron, $'O'$ is more than the MSE.

$$E_O(i) = \frac{1}{2}\sum_{p=1}^{P}[t_{po} - y_{po}(i)]^2 \geq \frac{E(i)}{O} \quad (10)$$

The purpose is to change the weights $w_{ko}(i+1)$ associated with this output $'O'$. The network weights will remain unchanged, causing the output to method the expectable output, allowing the network to escape the local minimum and accelerate the rate of convergence.

$$y_{PO}(i+1) = y_{PO}(i) + \beta\big(t_P(i) - y_{PO}(i)\big) = \beta t_{PO}(i) + (1-\beta)O_{PO}z \quad (11)$$

Where, $0 < \beta < 1$ for all data samples $'P'$.

The output is to obtain the following iteration's weights:

$$y_{pm}(i+1) = f\left(\sum_{k=1}^{K} w_{ko}(i+1)\,\bar{y}_{pk}(i+1)\right) \quad (12)$$

And thus,

$$\sum_{k=1}^{K}\Delta w_{ko}(i+1)\,\bar{y}_{pk}(i+1) = \varepsilon_{po} \quad (13)$$

Where,

$$\varepsilon_{po} = f^{-1}\left(y_{po}(i+1)\right) - \sum_{k=1}^{K} w_{ko}(i)\,\bar{y}_{pk}(i+1) \quad (14)$$

$$f^{-1}(x) = ln\left(\frac{x}{1-x}\right) \quad (15)$$

So, the weight is calculated as $\Delta w_{ko}(i+1)$ by using Eq. (4):

$$\Delta w_{ko}(i+1) = \frac{\sum_{P=1}^{P} e_{po}\,\bar{y}_{Pk}(i+1)}{\sum_{P=1}^{P}\bar{y}_{pk}^2(i+1)} \quad (16)$$

The weights and errors are calculated as follows:

$$E_o^{(k)}(i+1) = \frac{1}{2}\sum_{p=1}^{P}\sum_{m=1}^{M}\left[t_{po} - y_{po}(i+1)\right]^2 \quad (17)$$

An error greater than the average is selected, and the error is then reduced systematically by disturbing the result of the output layer. In the output neurons m, t_{pm} is the target, and the expectable output is $y_{PO}(i) + \beta\big(t_P(i) - y_{PO}(i)\big)$. Needless behavior will result if β is too near to 1. When β is set to 1, the target value will then be the probable output. In one cycle, it is problematic to train the structure to the desired output. Algorithm **1** shows the process of the DWM algorithm.

5. PROPOSED DWM+ELM

ELM is the way of learning that is the most quickly compared to certain conventional ANNs algorithms. However, randomly determining the input weights and hidden biases may reveal the variability in the ELM's prediction performance over iterations and indicate that the model is unstable. The ELM will become more stable, and its prediction performance will increase by taking into account the optimized weights and bias values. However, its weight modification method has a variety of limitations, including local optimum and low convergence rate. Hence, the weights between the output and hidden layers should be modified in a deterministic manner, according to the current study. The DWM algorithm is used to modify the connection weights and bias to reduce the system error and output value from deviating from predictions. DWM has improved global convergence and the rate of convergence of ELM.

DWM can increase ELM ability since it adjusts the ELM deterministically. An input with an error above the overall mean is chosen for the DWM between the hidden and input layers. The error is then significantly reduced by altering the error in a deterministic manner and then immediately reversing the learning to determine the weights in the input layer. The system error is anticipated to be close to the predefined one and smaller than the existing one. By adjusting the weights between the hidden and input layers, DWM adjusts the output value and system error to be close to the expected results. Algorithm **1** and Fig. (**1**) show the framework of the

Algorithm 1: Deterministic Weight Modification (DWM) algorithm

While (converged)
$\Delta E = [E(i) - E(i-t)] / t$
If $(\Delta E \leq G_T)$ and then
For all p
$E_p(i+1) = \frac{1}{2} \sum_{m=1}^{M} (t_{pm} - o_{pm}(i+1))]^2$
$p^* = \text{minimum}(E_p(i+1))$
End for
m^* is selected randomly between and 1
For all k
$k^* = \text{minimum}(\min(\bar{o}_{p^*k}(i), 1 - \bar{o}_{p^*k}(i))$
End for
If then
$o_{p^*m^*}(i+1) = 1 - \sqrt{2\lambda E(i)/PM}$
Else
$o_{p^*m^*}(i+1) = \sqrt{2\lambda E(i)/PM}$
End if
$\Delta\bar{o}_{p*k*}(i+1) = \frac{(f^{-1}(o_{p*m*}(i+1)) - \sum_{k=1}^{K} w_{km*}(i+1) o_{p*k*}(i))}{w_{k*m*}(i+1)}$
$\bar{o}_{p^*k^*}(i+1) = \Delta\bar{o}_{p^*k^*}(i+1) + \bar{o}_{p^*k^*}(i)$
$\varepsilon_{p^*k^*} = f^{-1}(\bar{o}_{p^*k^*}(i+1) - f^{-1}(\bar{o}_{p^*k^*}(i)$
For all n
$\Delta\bar{w}_{nk^*}(i+1) = \bar{\varepsilon}_{p^*k^*} \Big/ \sum_{n=1}^{N} x_{p^*n}^2$
End for
End if
End while

Algorithm 2: Step-by-step processes of the proposed frameworks

Step 1: Initializing the required parameters
Step 2: Collecting financial information from a data source
Step 3: The sample data is split into training sets and test sets after being normalized (using the min-max method)
Step 4: The DWM is employed in the beginning stages to determine the best input layer weight and hidden layer threshold from the training sample data (Algorithm 1)
Step 5: The number of best-hidden layer neurons is determined by trial and error.
Step 6: The output layer's weight is updated continuously

Note: (Fig. **1**).

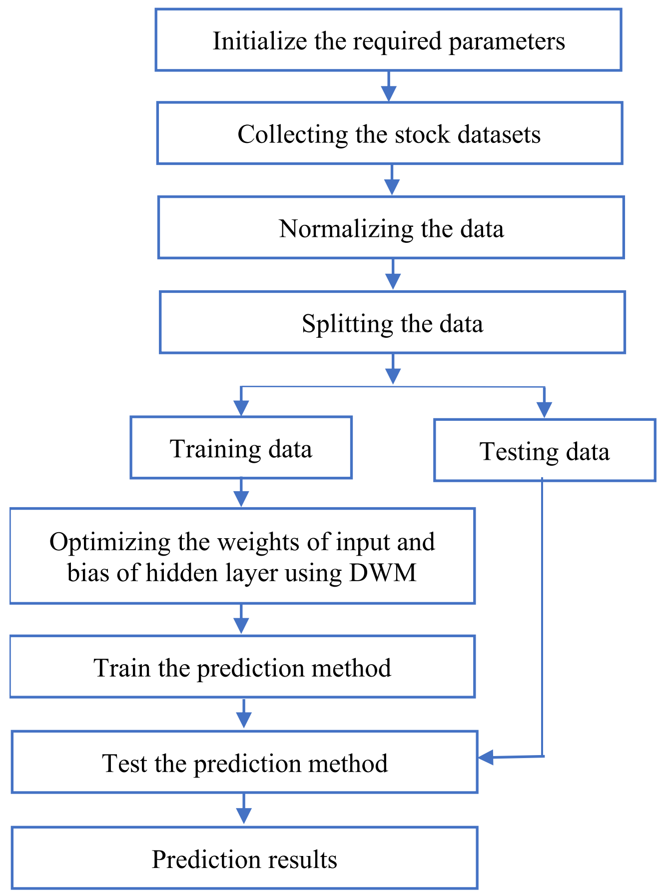

Fig. (1). Proposed frameworks for stock price prediction.

proposed DWM+ELM method. Our suggestion, known as DWM-ELM, is based on an a priori DWM approach that tries to increase (maximize) prediction accuracy over the training dataset while minimizing the number of hidden neurons. The same study has already been presented and published at a conference [31], and this article expands on that research. A Step-by-step of proposed frameworks is shown in Algorithm **2**.

6. RESULTS

This section evaluates the DWM+ELM's ability to estimate future prices using a variety of benchmark stock market datasets. The analysis findings are implemented using MATLAB R2015b. The DWM-ELM is compared with some of the variants of ELM algorithms, including a novel GA-based regularization online ELM called GA-ROSELM [16], optimized GA–ELM (OGA-ELM) [32], self-adaptive evolutionary ELM (SaE-ELM) [33], LM-ELM [34], ELM [35], and BPNN [36].

6.1. Datasets

The DWM+ELM was developed to predict the stock price. Two familiar Indian stock indices, such as Nifty 50 and S&P BSE Sensex and two Indian stock bank price datasets, namely State Bank of India (SBIN) and ICICI bank (ICICIBANK), were used for experiments. The datasets were obtained from Yahoo [37]. The datasets were collected between January, 2015 and December, 2022, which were divided into two phases, such as training and testing. Furthermore, 75% of the total datasets were used for training, and the remaining 25% were used for testing. The dataset included the day's opening, high, average, and closing prices. To employ neural networks to forecast future behavior, the gathered raw datasets were converted into several relevant financial technical indicators [38, 39]. Technical terms were found to have a unique inherent quality that enables traders to predict the closing price or index of stocks. To forecast stock prices, essential technical indicators were employed to extract the necessary evidence from the data. The objective value of the model was the closing price of the stock data on a particular transaction day. Scaling was carried out as follows:

$$X_i' = \left(\frac{X_i - X_{\min}}{X_{\max} - X_{\min}}\right)(X_{\max}' - X_{\min}') + X_i' \tag{18}$$

Where, X'_{max} and X'_{min} are the maximum and minimum values, respectively. $X_i, Xmin_{max}$ are present, maximum, and minimum real input values, respectively. The particulars of the technical indicator are defined in Table **1**.

6.2. Parameter Analysis

The architecture of ELM is the combination of different layers, such as input, hidden, and output layers. The neu-

Table 1. Technical indicators.

S. No.	Technical Indicators	Formulas	Descriptions
1	Simple Moving Average (SMV)	$MV = \frac{x_1 + x_2 + \cdots + x_n}{n}$	The average value of the number of days.
2	10-days Moving Average	$MV_{10} = \frac{x_1 + x_2 + \cdots + x_n}{n}$	The average value for the previous ten days.
3	Momentum	$M = C_t - C_{t-4}$	It measures the amount of stock price over a given period.
4	Stochastic K%	$STCK = \frac{C_t - LL_{t-n}}{HH_{t-n} - LL_{t-n}} \times 100$	Stochastic provides a benchmark for estimating the speed of price reversal. At a particular time, frame, the modern closing price's position is measured relative to other prices.
5	Stochastic D%	$STCD = \frac{\sum_{i=0}^{n-1} K_{t-1\%}}{n}$	$D\%$ stipulate the three days moving average of K%.
6	Relative Strength Index (RSI)	$= 100 - \frac{100}{1 + (\sum_{t=0}^{n-1} \mathrm{UP}_{t-1}/\mathrm{n})/(\sum_{t=0}^{n-1} \mathrm{DW}_{t=1}/\mathrm{n})}$	It is a movement oscillator that measures the rate and direction of price movement on a scale from 0 to 100.
7	Williams %R	$LW = \frac{H_n - C_t}{H_n - L_n} \times 100$	An opportunity to enter or exit the market is identified using this momentum indicator, which also calculates overbought and oversold levels.
8	Moving Average Convergence Divergence (MACD)	$= MACD(n)_{t-1} + \frac{2}{n+1}(Diff_t - MACD(n)_{t-1})$	The purpose of MACD is to predict a stock's future direction by matching up to its short- and long-term momentum.
9	Commodity Channel Index (CCI)	$CCI = \frac{M_t - SM_t}{0.015 D_t} \times 100$	CCI compares the current price level to the average price level over a certain period to identify new movements or signal dangerous situations.
10	Price Oscillator (PO)	$\mathrm{PO} = \frac{\mathrm{MA}_5 - \mathrm{MA}_{10}}{\mathrm{MA}_5}$	An essential indicator that displays the relationship between two moving averages.

C_t- Closing price; L_t- Lowest price; LL_t- Lowest Low; H_t- High price; HH_t-Highest high price; UP_t- Upward price; DW_t- Downward price.

rons of the input and output layers are described based on given applications. In the present article, ten technical indicators are considered as input neurons and one output neuron is considered closing price/index. The ELM algorithm has a single hidden layer, and the quantity of hidden layer neurons is crucial in the SLFN model. As few neurons will lead to under-fitting while too many will result in over-fitting, the prediction effect will be reduced. As a result, this study continuously tests and modifies the network's hidden layer node count to identify a better number of nodes. The ELM network's chosen maximum hidden layer neurons is 100, and the optimum number of hidden neurons is selected based on the trial-error method. When there are 36 hidden layer neurons, the learning model's MSE is the smallest.

The results are computed from an average of 20 independent runs, which is shown in Fig. (**2**). The activation function of ELM also has another important role in deciding prediction accuracy. Hence, the present article chose the sigmoid as the best activation function for ELM among four different activation functions, such as sigmoid, tanh, softsign, and ReLu. Its result is shown in Fig. (**3**). The termination condition is set to 0.0005. The maximum number of epochs is set to 1000. In the DWM algorithm, the two parameters also decide the prediction accuracy when selecting optimal values. Hence, the value of both is set as $\lambda = 0.2 \beta = 0.7$, based on the previous study [28].

6.3. Performance Measures

DWM+ELM performance is measured using different measures, namely Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Directional Symmetry (DS).

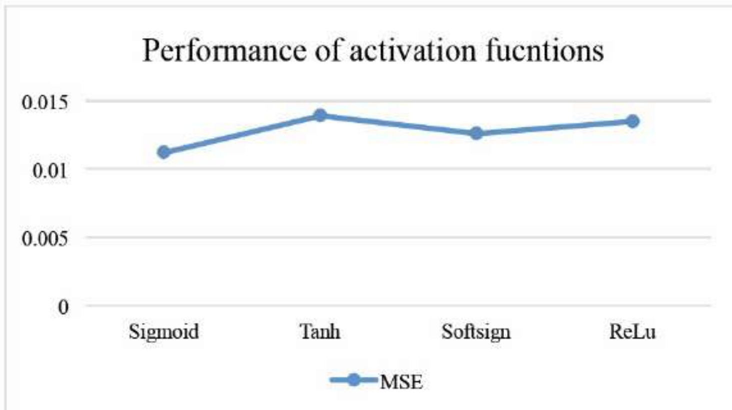

Fig. (2). Performance of activation functions. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

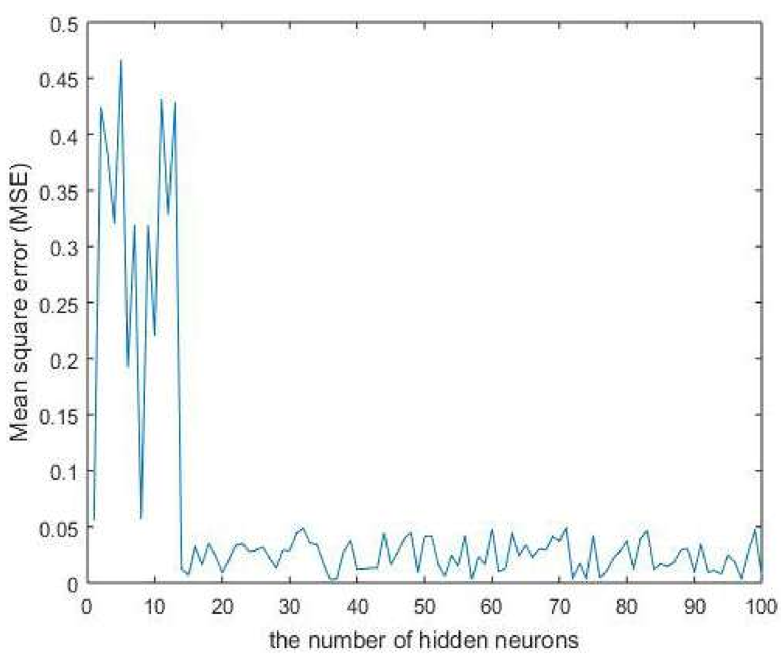

Fig. (3). Selection of best-hidden neuron in a hidden layer. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(t_{(i)} - y_{(i)}\right)^2} \quad (19)$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\left(t_{(i)} - y_{(i)}\right)\right| \quad (20)$$

$$MAPE = \sqrt{\frac{100}{N}\sum_{i=1}^{N}\left|\frac{\left(t_{(i)} - y_{(i)}\right)}{t_{(i)}}\right|} \quad (21)$$

$$DS = \frac{100}{N}\sum_{i=1}^{N} d_i\, , d_i = \begin{cases} 1 (t_{(i)} - t_{(i-1)})(y_{(i)} - y_{(i-1)}) \\ 0 otherwise \end{cases} \quad (22)$$

Where, $t_{(i)}$ is the target value, $y_{(i)}$ is the predicted value, and $'N'$ is the total number of samples.

7. DISCUSSION

The primary data was gathered from a financial website that has basic attributes that were then transformed into technical indicators. The presented prediction methods for predicting stock price consider the technical indicators as inputs. The results of comparing the prediction algorithms for the Nifty 50 stock indices, S&P Sensex indices, SBIN datasets, and ICICI datasets are shown in Tables **2-5**, respectively. Similar to this,

Table 2. Experimental results of the Nifty 50 Stock Indices.

Methods	RMSE	MAE	MAPE	DS
DWM+ELM	0.0096	0.0563	1.7045	89.34
GA+ROSELM	0.0102	0.0649	1.9503	87.81
SaE-ELM	0.0124	0.0739	2.3961	85.23
LM-ELM	0.0316	0.0792	3.0693	81.84
ELM	0.0139	0.0896	4.7291	78.34
BPNN	0.0158	0.0949	5.0718	72.89

Table 3. Experimental of the S&P BSE Sensex Stock Indices.

Methods	RMSE	MAE	MAPE	DS
DWM+ELM	0.0485	0.0391	1.4726	92.42
GA+ROSELM	0.0658	0.0467	2.8710	87.19
SaE+ELM	0.0726	0.0574	2.2826	82.26
LM+ELM	0.0816	0.0659	4.0372	78.92
ELM	0.0893	0.0736	5.2717	73.89
BPNN	0.0975	0.0844	6.7225	64.36

Table 4. Experimental results of the SBIN.

Methods	RMSE	MAE	MAPE	DS
DWM+ELM	0.0107	0.0069	1.2562	91.42
GA+ROSELM	0.0112	0.0075	1.9372	89.78
SaE+ELM	0.0118	0.0079	2.3810	85.91
LM+ELM	0.0251	0.0082	3.4821	81.03
ELM	0.0273	0.0086	4.8026	75.10
BPNN	0.0319	0.0092	5.1729	68.29

Table 5. Experimental results of the ICICI.

Methods	RMSE	MAE	MAPE	DS
DWM+ELM	0.0102	0.0061	1.5924	88.45
GA+ROSELM	0.0127	0.0064	1.8274	86.93
SaE+ELM	0.0192	0.0071	2.5901	83.67
LM+ELM	0.0218	0.0078	3.7392	80.91
ELM	0.0248	0.0085	4.0182	63.02
BPNN	0.0371	0.0092	4.9271	55.91

the graphs in Figs. (**4-7**) show the performance comparisons for the Nifty 50, S&P Sensex, SBIN, and ICICI datasets, respectively. Four variant ELM algorithms and BPNN algorithms were taken for numerical values compared to the proposed method. However, for a better understanding of prediction methods, three recently developed prediction methods were considered, such as GA+ROSELM, OGA+ELM, and SaE-ELM in graphical representation.

From the experimental results, the DWM+ELM method produced higher prediction accuracy and faster convergence rate than other existing algorithms. As mentioned in Table **2**, the RMSE value for the suggested DWM+ ELM technique was extremely low, *i.e*., 0.0096. The comparative algorithms GA+ROSELM, SaE-ELM, LM-ELM, ELM, and BPNN generated the values of 0.0102, 0.0124, 0.0316, 0.0139, and 0.0158, respectively. The suggested DWM+ELM technique's MAE value was a low 0.0563. The values 0.0649, 0.0739, 0.0792, 0.0896, and 0.0949 were produced by comparing algorithms GA+ROSELM, SaE-ELM, LM-ELM, ELM, and BPNN, respectively. The suggested DWM+ELM technique's

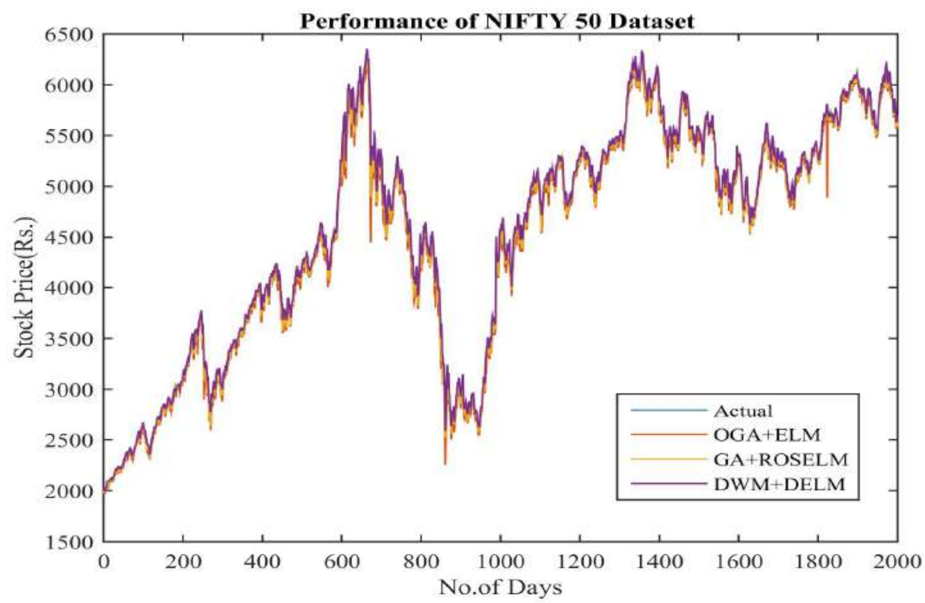

Fig. (4). Performance results for Nifty 50 datasets. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

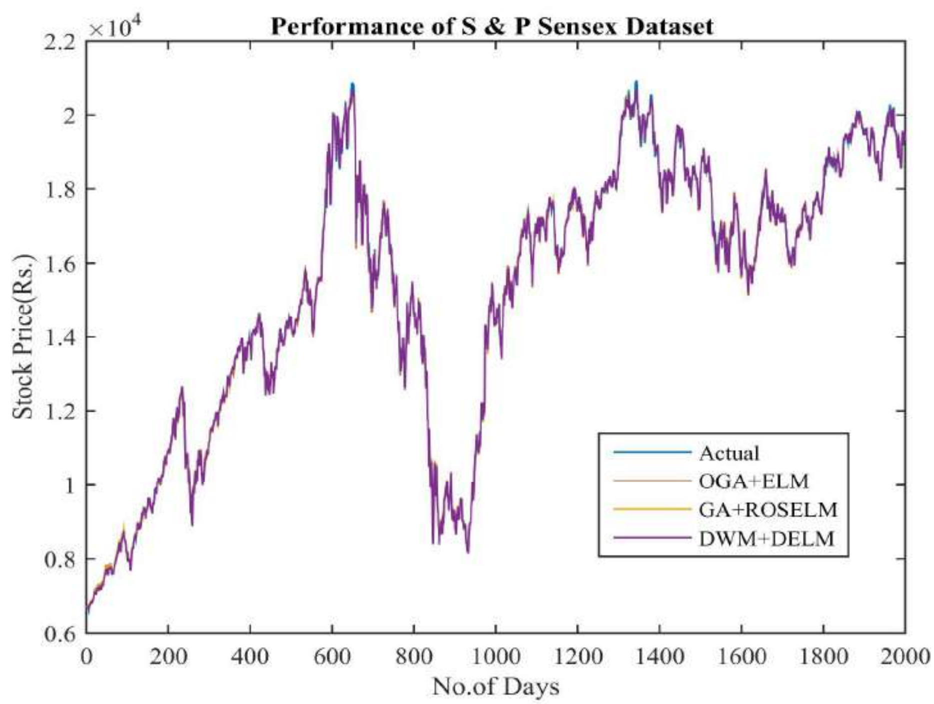

Fig. (5). Performance results of S & P Sensex datasets. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

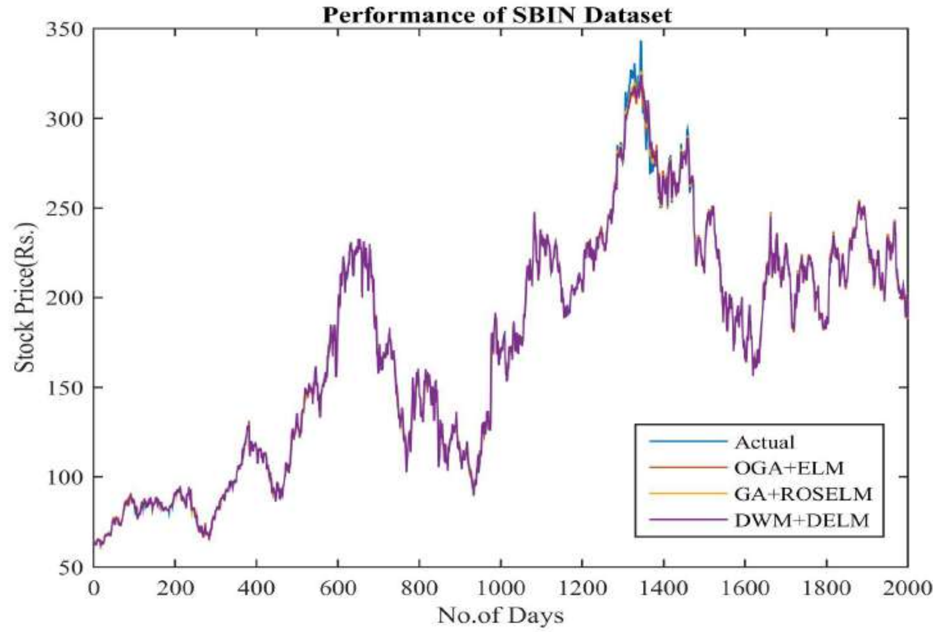

Fig. (6). Performance results of the SBIN dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

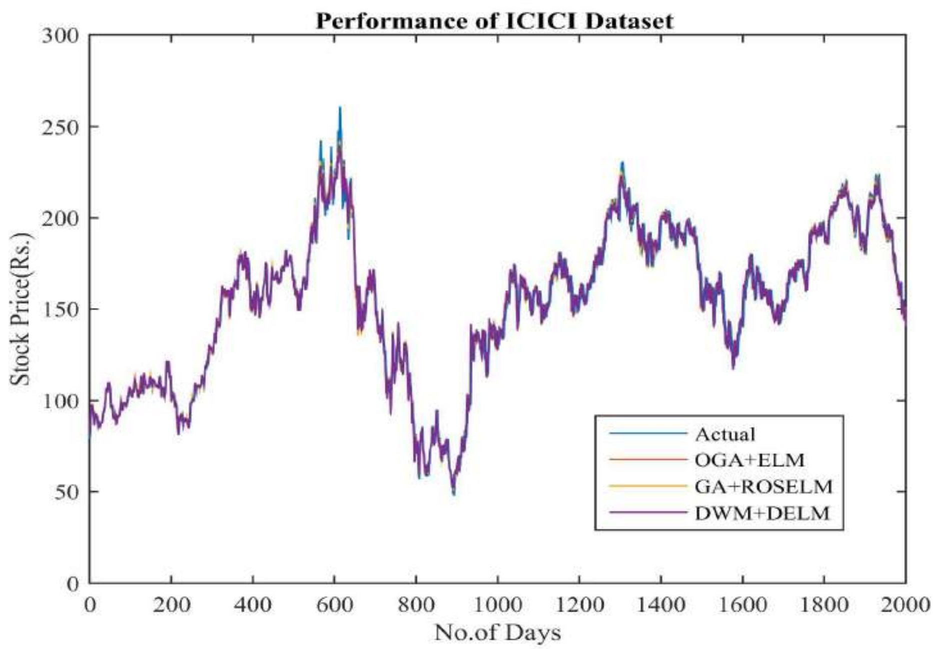

Fig. (7). Performance results of ICICI datasets. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

MAPE value was a low 1.7045. The values 1.9503, 2.3961, 3.0693, 4.7291, and 5.0718 were produced by comparing algorithms GA+ROSELM, SaE-ELM, LM-ELM, ELM, and BPNN, respectively. The suggested DWM+ELM technique's DS value was very high that was 89.34. The values 87.81, 85.23, 81.84, 78.34, and 72.89 were produced by the comparing algorithms GA+ROSELM, SaE-ELM, LM-ELM, ELM, and BPNN, respectively. Similarly, from Tables **3-5**, the proposed DWM+ELM was produced with high accuracy. Similar to what was observed in earlier tests, DWM-ELM was able to attain accuracy levels that were on par with compared methods while significantly reducing the number of hidden neurons. On the other side, an increase in training time could be observed. The accuracies were comparable when comparing DWM-ELM with the recent variant ELM and BPNN. However, DWM-ELM had a faster training process and fewer hidden neurons. According to the experimental findings, the suggested DWM+ELM consistently outperformed existing prediction methods when accuracy and convergence were taken into account. It is a good learning method, and it is exactly what the customers or investors in the real trading market need.

CONCLUSION

The DWM+ELM learning model is proposed as a new learning model for stock market prediction. The DWM algorithm was used in the suggested method to optimize the

ELM and improve prediction accuracy and convergence rate. DWM-ELM eliminated the unpredictability of the original ELM by evaluating input weights and biases rather than randomly assigning them. In terms of RMSE, MAE, MAPE, and DS, the performance was compared with recently suggested different variants of the ELM algorithm and the BPNN. Two stock indices and two stock market bank datasets were used in the experiment. The experimental finding showed that the suggested DWM+ELM technique achieved higher prediction speed and accuracy. In the future, instead of depending simply on price data, various market information sources, such as news and blogs, can be used to extract public views to increase prediction accuracy.

DWM is a specific way of updating the model weights without introducing randomness or adaptive learning rates during training. However, DWM has the disadvantage of occasionally being sensitive to the network's starting weights. DWM might not be able to converge to a satisfactory solution if the initial weights are not selected appropriately. DWM also has the potential to be computationally costly. This can make the software implementation of DWM challenging. On the other hand, swarm intelligence (SI) based-optimization techniques encompass a broader range of algorithms and strategies that aim to find optimal weight values by iteratively adjusting the parameters based on the gradients and possibly introducing randomness, adaptive learning rates, and momentum to improve convergence and overall training performance.

CURRENT & FUTURE DEVELOPMENTS

Even though ELM is recognized for having a faster training pace than standard neural networks, computational difficulties may still arise from very big datasets or overly complicated structures. On the other hand, ELM may handle outliers poorly by nature. Extreme occurrences can occur in the financial markets, and if outliers are not properly handled, they can have a substantial negative effect on the model's performance. On the other hand, time series data may be effectively modeled with Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs) to represent sequential and temporal relationships. Over time, patterns may be seen in stock values, and deep learning models are better than ELM in capturing these trends. Abstract and hierarchical characteristics are automatically learned from raw data using deep learning algorithms. This can be useful for stock market prediction as it eliminates the need for laborious human feature engineering since deep learning models can extract pertinent features from the input time series data.

LIST OF ABBREVIATIONS

ANN	=	Artificial Neural Network
ELM	=	Extreme Learning Machine
ML	=	Machine Learning
SLFN	=	Single-hidden Layer Neural Network
SI	=	Swarm Intelligence

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data and supportive information are available within the article.

FUNDING

None.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] G.E. Box, G.M. Jenkins, G.C. Reinsel, and G.M. Ljung, *Time series analysis: Forecasting and control..* John Wiley & Sons, 2015.

[2] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques", *Expert Syst. Appl.,* vol. 42, no. 1, pp. 259-268, 2015. http://dx.doi.org/10.1016/j.eswa.2014.07.040

[3] S. Asadi, E. Hadavandi, F. Mehmanpazir, and M.M. Nakhostin, "Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction", *Knowl. Base. Syst.,* vol. 35, pp. 245-258, 2012. http://dx.doi.org/10.1016/j.knosys.2012.05.003

[4] K. Kalaiselvi, K. Velusamy, and C. Gomathi, "Financial prediction using back propagation neural networks with opposition based learning", *J. Phys.: Conf. Ser.,* vol. 1142, p. 012008, 2018. http://dx.doi.org/10.1088/1742-6596/1142/1/012008

[5] V.K. David, "A survey on ontology based clinical practice guidelines with soft computing techniques", *Int. J. Pure Appl. Math.,* vol. 118, no. 18, pp. 3047-3051, 2018.

[6] C. Anand, "Comparison of stock price prediction models using pre-trained neural networks", *J. Ubiquitous Computing Commun. Technol.,* vol. 3, no. 2, pp. 122-134, 2021. http://dx.doi.org/10.36548/jucct.2021.2.005

[7] H.K. Andi, "An accurate bitcoin price prediction using logistic regression with LSTM machine learning model", *J. Soft Computing Paradigm,* vol. 3, no. 3, pp. 205-217, 2021. http://dx.doi.org/10.36548/jscp.2021.3.006

[8] V.K. David, and S. Rajasekaran, *Pattern recognition using neural and functional networks..* Springer, 2009. http://dx.doi.org/10.1007/978-3-540-85130-1

[9] C.L. Lin, J.F. Wang, C.Y. Chen, C.W. Chen, and C.W. Yen, "Improving the generalization performance of RBF neural networks using a linear regression technique", *Expert Syst. Appl.,* vol. 36, no. 10, pp. 12049-12053, 2009. http://dx.doi.org/10.1016/j.eswa.2009.03.012

[10] S. Rajasekaran, and R. Amalraj, "Predictions of design parameters in civil engineering problems using SLNN with a single hidden RBF neuron", *Comput. Struc.,* vol. 80, no. 31, pp. 2495-2505, 2002. http://dx.doi.org/10.1016/S0045-7949(02)00213-4

[11] K. Velusamy, and R. Amalraj, "Performance of the cascade correlation neural network for predicting the stock price", *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)* Coimbatore, India, 2017, pp. 1-6. http://dx.doi.org/10.1109/ICECCT.2017.8117824

[12] K. Velusamy, and R. Amalraj, "Prediction of stock market using cascade correlation neural network with principal component analysis", *Int. J. Eng. Technol.,* vol. 7, pp. 3485-3488, 2018.

[13] X. Li, H. Xie, R. Wang, Y. Cai, J. Cao, F. Wang, H. Min, and X. Deng, "Empirical analysis: Stock market prediction *via* extreme learning machine", *Neural Comput. Appl.,* vol. 27, no. 1, pp. 67-78, 2016. http://dx.doi.org/10.1007/s00521-014-1550-z

[14] M. Jiang, L. Jia, Z. Chen, and W. Chen, "The two-stage machine learning ensemble models for stock price prediction by combining

mode decomposition, extreme learning machine and improved harmony search algorithm", *Ann. Oper. Res.,* vol. 309, no. 2, pp. 553-585, 2022.
http://dx.doi.org/10.1007/s10479-020-03690-w

[15] A.S.C. Alencar, A.R. Rocha Neto, and J.P.P. Gomes, "A new pruning method for extreme learning machines *via* genetic algorithms", *Appl. Soft Comput.,* vol. 44, pp. 101-107, 2016.
http://dx.doi.org/10.1016/j.asoc.2016.03.019

[16] F. Weng, Y. Chen, Z. Wang, M. Hou, J. Luo, and Z. Tian, "Gold price forecasting research based on an improved online extreme learning machine algorithm", *J. Ambient Intell. Humaniz. Comput.,* vol. 11, no. 10, pp. 4101-4111, 2020.
http://dx.doi.org/10.1007/s12652-020-01682-z

[17] L.L. Li, Z.F. Liu, M.L. Tseng, K. Jantarakolica, and M.K. Lim, "Using enhanced crow search algorithm optimization-extreme learning machine model to forecast short-term wind power", *Expert Syst. Appl.,* vol. 184, p. 115579, 2021.
http://dx.doi.org/10.1016/j.eswa.2021.115579

[18] S. Das, T.P. Sahu, R.R. Janghel, and B.K. Sahu, "Effective forecasting of stock market price by using extreme learning machine optimized by PSO-based group oriented crow search algorithm", *Neural Comput. Appl.,* vol. 34, no. 1, pp. 555-591, 2022.
http://dx.doi.org/10.1007/s00521-021-06403-x PMID: 34413575

[19] P. Yang, D. Wang, W.B. Zhao, L.H. Fu, J.L. Du, and H. Su, "Ensemble of kernel extreme learning machine based random forest classifiers for automatic heartbeat classification", *Biomed. Signal Process. Control,* vol. 63, p. 102138, 2021.
http://dx.doi.org/10.1016/j.bspc.2020.102138

[20] Chandar, "Grey wolf optimization-elman neural network model for stock price prediction", *Soft Comput.,* vol. 25, no. 1, pp. 649-658, 2021.
http://dx.doi.org/10.1007/s00500-020-05174-2

[21] D. Selvamuthu, V. Kumar, and A. Mishra, "Indian stock market prediction using artificial neural networks on tick data", *Financial Innovation,* vol. 5, no. 1, p. 16, 2019.
http://dx.doi.org/10.1186/s40854-019-0131-7

[22] D. Wu, X. Wang, and S. Wu, "A hybrid method based on extreme learning machine and wavelet transform denoising for stock prediction", *Entropy,* vol. 23, no. 4, p. 440, 2021.
http://dx.doi.org/10.3390/e23040440 PMID: 33918679

[23] Z. Tang, T. Zhang, J. Wu, X. Du, and K. Chen, "Multistep-ahead stock price forecasting based on secondary decomposition technique and extreme learning machine optimized by the differential evolution algorithm", *Math. Probl. Eng.,* vol. 2020, 2020.

[24] P. Khuwaja, S.A. Khowaja, I. Khoso, and I.A. Lashari, "Prediction of stock movement using phase space reconstruction and extreme learning machines", *J. Exp. Theor. Artif. Intell.,* vol. 32, no. 1, pp. 59-79, 2020.
http://dx.doi.org/10.1080/0952813X.2019.1620870

[25] Z. Cui, B. Huang, H. Dou, Y. Cheng, J. Guan, and T. Zhou, "A two-stage hybrid extreme learning model for short-term traffic flow forecasting", *Mathematics,* vol. 10, no. 12, p. 2087, 2022.
http://dx.doi.org/10.3390/math10122087

[26] P. Chen, A. Vivian, and C. Ye, "Forecasting carbon futures price: A hybrid method incorporating fuzzy entropy and extreme learning machine", *Ann. Oper. Res.,* vol. 313, no. 1, pp. 559-601, 2022.
http://dx.doi.org/10.1007/s10479-021-04406-4 PMID: 35002000

[27] G.B. Huang, Q.Y. Zhu, and C.K. Siew, "Extreme learning machine: Theory and applications", *Neurocomputing,* vol. 70, no. 1-3, pp. 489-501, 2006.
http://dx.doi.org/10.1016/j.neucom.2005.12.126

[28] S.C. Ng, C.C. Cheung, and S.H. Leung, "Magnified gradient function with deterministic weight modification in adaptive learning", *IEEE Trans. Neural Netw.,* vol. 15, no. 6, pp. 1411-1423, 2004.
http://dx.doi.org/10.1109/TNN.2004.836237 PMID: 15565769

[29] C. Wang, and D.J. Hill, *Deterministic learning theory for identification, recognition, and control..* CRC Press, 2009.

[30] K. Velusamy, and R. Amalraj, "Cascade correlation neural network with deterministic weight modification for predicting stock market price", *IOP Conf. Ser.: Mater. Sci. Eng.,* vol. 1110, 2021p. 012005
http://dx.doi.org/10.1088/1757-899X/1110/1/012005

[31] K. Kalaiselvi, and V.K. David, "Enhanced extreme learning machine algorithm with deterministic weight modification for investment decision on Indian stocks", *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)* Trichy, India, 2022, pp. 1409-1414.
http://dx.doi.org/10.1109/ICOSEC54921.2022.9951899

[32] M.A.A. Albadr, S. Tiun, M. Ayob, and F.T. AL-Dhief, "Spoken language identification based on optimised genetic algorithm–extreme learning machine approach", *Int. J. Speech Technol.,* vol. 22, no. 3, pp. 711-727, 2019.
http://dx.doi.org/10.1007/s10772-019-09621-w

[33] J. Cao, Z. Lin, and G.B. Huang, "Self-adaptive evolutionary extreme learning machine", *Neural Process. Lett.,* vol. 36, no. 3, pp. 285-305, 2012.
http://dx.doi.org/10.1007/s11063-012-9236-y

[34] X. Wang, and M. Han, "Improved extreme learning machine for multivariate time series online sequential prediction", *Eng. Appl. Artif. Intell.,* vol. 40, pp. 28-36, 2015.
http://dx.doi.org/10.1016/j.engappai.2014.12.013

[35] F. Zhang, "Extreme learning machine for stock price prediction", *Int. J. Electr. Eng. Educ.,* vol. •••, 2021.
http://dx.doi.org/10.1177/0020720920984675

[36] J.Z. Wang, J.J. Wang, Z.G. Zhang, and S.P. Guo, "Forecasting stock indices with back propagation neural network", *Expert Syst. Appl.,* vol. 38, no. 11, pp. 14346-14355, 2011.
http://dx.doi.org/10.1016/j.eswa.2011.04.222

[37] "Yahoo finance", Available From: https://finance.yahoo.com/

[38] Y. Kara, M. Acar Boyacioglu, and Ö.K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange", *Expert Syst. Appl.,* vol. 38, no. 5, pp. 5311-5319, 2011.
http://dx.doi.org/10.1016/j.eswa.2010.10.027

[39] Y. Zhang, and L. Wu, "Stock market prediction of S&P 500 *via* combination of improved BCO approach and BP neural network", *Expert Syst. Appl.,* vol. 36, no. 5, pp. 8849-8854, 2009.
http://dx.doi.org/10.1016/j.eswa.2008.11.028