American Journal of Business and Operations Research

ISSN: 2692-2967 Vol: 11, No. 02, 2024 PP. 44–50

Article DOI: https://doi.org/10.54216/AJBOR.110205



FULL LENGTH ARTICLE

Optimized Cash Forecasting Models for Banking Applications Using Soft Computing Techniques

Manjeet Rege[1,*]

[1] Professor, University of St. Thomas, Minnesota, USA

Email: rege@stthomas.edu

Received: November 14, 2023 Revised: January 11, 2024 Accepted: June 28, 2024 ⋆ Corresponding author

ABSTRACT

The Artificial Neural Network-based Cash Forecasting Model (ANN-CFM) is proposed to address limitations of linear approaches to financial management, particularly when data exhibit complex nonlinear behaviour. The model combines linear and nonlinear information using a feed-forward neural network with input, hidden, and output layers. Its short- and long-term forecasts account for working days, salary effects, holidays, month, and prior-year cash data. Backpropagation adjusts weights and biases to reduce prediction error, while Particle Swarm Optimization (PSO) is used to improve parameter optimization. The model is designed to process large datasets and adapt to changing financial conditions. The study reports an overall accuracy of approximately 95% compared with normal data and presents comparisons of cash-flow forecasts and error measures for short- and long-term prediction.

Keywords: ANN-CFM ▪ Artificial Neural Network ▪ Particle Swarm Optimization ▪ Cash forecasting ▪ Banking

1. INTRODUCTION

Cash management is essential to banking operations, where financial institutions must maintain sufficient liquidity while limiting the cost of holding excess cash. Traditional forecasting approaches based on linear relationships may not represent the nonlinear interactions among working days, holidays, salary payments, seasonal behaviour, and other factors that influence cash demand. Artificial Neural Network-based Cash Forecasting Models (ANN-CFM) offer an alternative by using neural computation to model these relationships and process large samples in parallel [1, 2].

The ANN-CFM uses a feed-forward neural-network architecture in which information passes from the input layer through hidden processing units to an output layer. The initial weights and biases are randomized and then adjusted during learning to identify cash requirements. Parallel processing allows the model to analyze large quantities of data, while the layered structure makes it possible to learn both linear and nonlinear relationships that may be difficult to express in conventional financial models.

An ANN typically consists of input, hidden, and output layers. Connections between neurons are weighted, and iterative learning algorithms adjust the weights and biases to reduce prediction error. ANN-CFM can support both short- and long-term forecasts: short-term estimates may account for the average number of working days, holidays, and salary dates, while longer-term forecasts may use the month and reference year. Backpropagation updates model parameters from the difference between predicted and observed cash demand, enabling the model to adapt as new data become available [3].

This feedback process begins with initial weights and biases and repeatedly corrects them to reduce the difference between forecast and actual cash demand. The network can include additional layers and neurons for datasets of different complexity. Such adaptive learning is intended to make forecasting less dependent on fixed, sequential assumptions and

more capable of responding to nonlinear changes in financial activity.

The ability to learn nonlinear patterns is useful when cash demand is affected by unexpected events or interactions among several factors. Unlike conventional linear regression and time-series approaches, neural networks can represent more complex relationships and may therefore provide banks with more informative cash-flow projections [4, 5]. Data normalization, careful selection of input parameters, and an appropriate network structure are also important to make the model suitable for practical financial scenarios. Backpropagation and gradient-based optimization can further reduce forecast errors [6, 7].

The model is intended to combine the benefits of established forecasting methods with the computational demands of modern financial operations. Its ability to process substantial data volumes, learn from correlations, and adjust to changing conditions is presented as a way to reduce approximation errors over repeated training cycles. Maintaining this flexibility is particularly important when internal and external factors cause cash requirements to vary over time.

Cash demand is influenced by historical patterns as well as seasonal factors, including holidays and salary dates. By incorporating these dynamic variables and applying continuous training, ANN-CFM is intended to improve estimates of liquidity needs and reduce the risk of both cash shortages and excess balances [8]. Its ability to process large volumes of data and adapt to changing market and customer behaviour can support more flexible cash-management decisions [9, 10].

The model is also presented as suitable for integration with existing banking systems. Its modular inputs can be adjusted as an institution's financial parameters and cash-management requirements change, supporting use in both large commercial banks and smaller community banks. In this way, cash forecasting and allocation could move toward more adaptive, digitally oriented processes with the potential to reduce operating costs and inefficient financing risks.

2. RELATED WORK

Prior work has explored artificial neural networks, feedforward networks, and optimization methods for problems involving nonlinear data and forecasting. Chaos theory describes systems that may exhibit nonlinearity, ergodicity, determinism, and sensitivity to initial conditions; its search properties have motivated applications in engineering, information science, biology, and optimization. Chaotic mappings have been proposed to improve PSO by replacing conventional pseudorandom sequences, with the aim of reducing premature convergence. Genetic algorithms and PSO are both used for optimization, but PSO particles retain information about their own and neighboring search results, whereas genetic algorithms do not use the same particle-memory mechanism [11].

PSO is used to search for promising candidate solutions through cooperation among particles. In financial applications, combining this global search with an ANN's local learning is intended to find better model parameters than either process alone. The cited literature also discusses feedforward networks, backpropagation, normalization, short- and long-term forecasting inputs, and hybrid ANN–PSO designs.

Table 1 summarizes methods discussed in the source study, together with their reported advantages and limitations. The literature review identifies a gap in cash forecasting: conventional statistical methods may omit holiday effects and day-of-month information, and modern soft-computing approaches have not been applied consistently to both short- and long-term cash predictions.

3. OBJECTIVE OF THE RESEARCH WORK

The main objective is to develop a reliable and effective ANN-CFM system that overcomes limitations of standard linear models and predicts financial institutions' cash requirements. The study uses neural computation to analyze linear and nonlinear data and account for working days, salary disbursements, holidays, and seasonal changes. By combining feedforward neural networks, backpropagation, and ANN–PSO optimization, the proposed approach aims to reduce prediction error and operational risk while improving the speed and flexibility of cash forecasting.

4. MOTIVATION FOR THE RESEARCH WORK

Simple deterministic techniques can be useful when demand for cash and other financial assets follows a linear pattern. In practice, cash demand often behaves as a nonlinear stochastic process. Salary distributions, holidays, and daily fluctuations may therefore be inadequately represented by basic regression or time-series models. The need for accurate and timely estimates motivates a method that can process large datasets, model nonlinear relationships, and optimize cash projections. ANN-CFM applies neural computation and PSO to support efficient use of resources and reduce liquidity risk in a changing financial environment.

5. THE PROJECTED METHOD

Developing a neural network for a complex forecasting problem requires careful selection of computational variables and model structure. The procedure described in the study defines and categorizes input variables, normalizes bank data, selects short- and long-term inputs, chooses the number of hidden neurons and network layers, specifies the activation function, divides observations into training and testing sets, trains the network, and evaluates the optimized results.

The ANN-CFM architecture has three layers: input, hidden, and output. The input layer captures the factors that influence cash demand: Holiday Effect (HDE), Weekday of the Month (WDOM), Working Days Offset (WDO), Salary Day Effect (SDE), Month of the Year (MOY), and cash data from the prior year (RY). Weekends and holidays may alter transaction patterns, while salary distributions can cause predictable increases in cash withdrawals. The combined inputs provide the basis for the forecast.

The hidden layer assigns weights and biases to the inputs and applies activation functions to identify nonlinear relationships.

Table 1. Summary of related works discovered

Technique	Advantages	Limitations
Feed-forward neural networks (FFNN) [12]	Direct data flow from input to output simplifies the design and works well with static data patterns.	Optimization may become trapped in local minima, and flexibility can be limited in dynamic settings.
Backpropagation for weight optimization [13]	Processes nonlinear data and reduces error through iterative updates to weights and biases.	Convergence may be slow for complex networks, and large datasets can lead to overfitting.
Short-term forecasting parameters [14]	Day-specific factors such as working days, salary effects, and holidays can improve accuracy.	External macroeconomic variables and broader trends may be omitted.
Long-term forecasting parameters [15]	Yearly and monthly trends support seasonal adjustment and longer-horizon accuracy.	Sudden economic changes or shifts in customer preferences may not be captured.
Hybrid ANN–PSO [16]	Combines ANN local learning with PSO global search to improve convergence and accuracy.	Computational requirements are high, and performance depends on parameter tuning.
Input-data normalization [17]	Stabilizes training by bringing input variables to consistent ranges.	Does not necessarily remove noise or resolve class and data imbalance.
ANN with a single hidden layer [18]	Models moderately complex relationships with relatively low computational cost.	May not scale to multidimensional data or capture subtle patterns in highly complex settings.
Chaotic PSO integration [19]	Chaotic search patterns can reduce premature convergence to local optima.	Selecting and tuning a suitable chaotic function requires dataset-specific investigation.
Sigmoid activation functions (Tansig and Logsig) [20]	Provide smooth gradients and nonlinear transformations used in financial forecasting.	Can perform poorly on sparse or discontinuous data and may suffer vanishing gradients in deep networks.
Real-time ANN-CFM processing [21]	Supports rapid response to changing cash-flow requirements and decisions.	Data standardization, preprocessing, and integration with live systems may introduce delays.
Large training datasets (e.g., 80/20 split) [22]	Diverse training patterns can improve generalization and prediction accuracy.	Long-term high-quality data can be difficult to obtain; noise and missing values affect accuracy.
Multi-layer perceptron (MLP) [23]	Models complex nonlinear relationships and permits flexible choices of hidden layers and neurons.	Insufficient training data can cause overfitting, and deeper networks increase computation.
Heuristic-based neural-network design [24]	Heuristics can address complex optimization problems by simulating decision processes.	Methods are not standardized, so performance can vary between datasets and budgets.
Evolutionary optimization (e.g., ANN–PSO) [25]	Expands the search space and can address local-minimum problems common to gradient methods.	Parameter tuning is computationally demanding, especially for large or rapidly changing datasets.
Holiday and festive-day effects	Seasonal inputs can improve forecasts when demand changes around holidays.	New events and geographic or cultural variation may be difficult to anticipate.
Hybrid evolutionary ANN training	Combining optimization strategies can improve convergence and reduce forecast error.	Hybrid methods require computational resources and careful tuning.
Dynamic PSO weight adjustment	Dynamic updates can help the model adapt to real-time data changes.	Results can be sensitive to initial parameters and may be unstable in changing settings.
Incremental ANN-CFM learning	The model can adapt to continuing data streams without restarting training.	Incremental updates may affect forecast precision and increase computation on large datasets.
Data preprocessing and normalization	Consistent inputs can speed convergence and reduce discrepancies during training.	Selecting preprocessing methods requires care to avoid removing useful information.
Salary-day effects in short-term forecasts	Captures predictable cash-demand increases around salary disbursements.	Applicability is limited where salary patterns are irregular or vary across industries.
ANN with real-time monitoring	Enables timely response to demand changes and supports efficient operations.	Integration can involve setup costs, data latency, and inaccurate live feeds.
Evaluation with MAE, MAPE, and MSE	Standardized metrics support comparison and iterative model improvement.	Metrics do not capture every aspect of model quality, such as interpretability or resilience.
Advanced activations (ReLU, Leaky ReLU)	Can improve deep-network training and reduce vanishing-gradient effects.	Poor tuning may introduce sparsity or limit performance on some nonlinear datasets.
Non-financial inputs (e.g., weather, public events)	Additional context may improve forecasts when external events affect demand.	Requires access to diverse data and can make models more complex and difficult to interpret.

Its computation is represented by

$$W = \sigma\left(\sum_{j=1}^{m} z_j y_j + c\right). \tag{1}$$

Here, y_j denotes an input such as HDE or WDOM, z_j is its coefficient, c is the bias, and σ is the activation function. A sigmoid activation is defined as

$$S = \sigma(y) = \frac{1}{1+e^{-y}}. \tag{2}$$

The output layer combines hidden-layer values and its own weights and bias to produce predicted cash demand:

$$d = \sum_{i=1}^{n} z_i W_i + c. \tag{3}$$

The value d is compared with actual demand for the corresponding period. During training, the weights and biases are updated to reduce the difference between predicted and observed outcomes. The weight update is written as

$$z_{m,n}^{(v+1)} = z_{m,n}^{(v)} - \frac{\partial K}{\partial z_{m,n}}, \tag{4}$$

where $z_{m,n}^{(v)}$ connects the mth neuron in the input layer to the

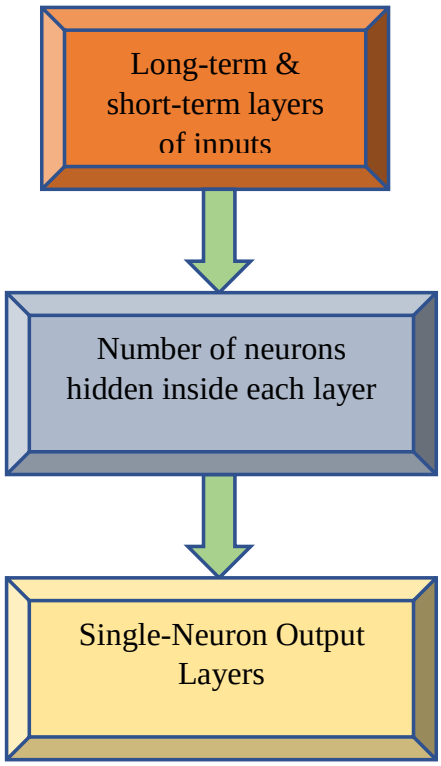


Figure 1. An overview of the ANN design.

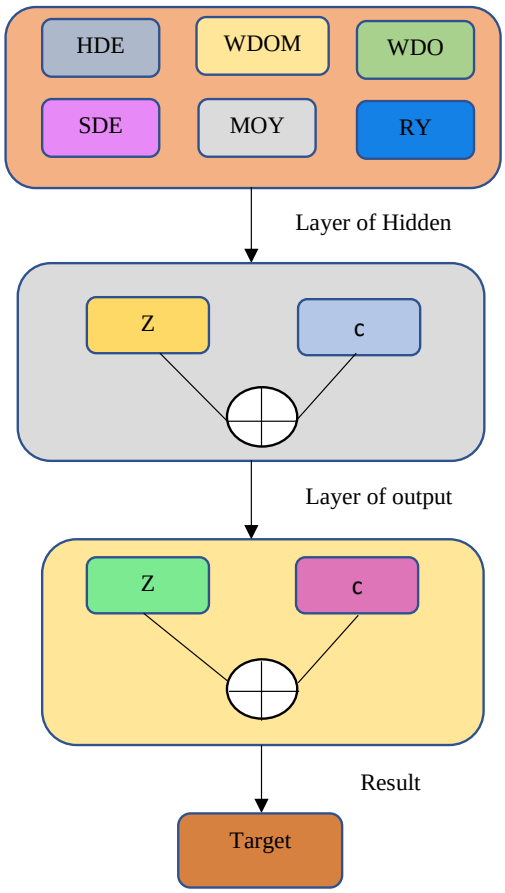


Figure 2. Framework of ANN-CFM.

nth hidden neuron at iteration v, and K is the loss function. Biases are updated similarly:

$$c_n^{(v+1)} = c_n^{(v)} - \frac{\partial K}{\partial c_n}. \tag{5}$$

The model also uses PSO to tune initial weights and biases. Particles search for good solutions by changing their positions according to individual and collective results. The velocity and position updates are

$$u_j^{(v+1)} = \omega u_j^{(v)} + d_1 s_1 \left(q_j^{\text{best}} - y_j^{(v)}\right) + d_2 s_2 \left(g^{\text{best}} - y_j^{(v)}\right), \tag{6}$$

$$y_j^{(v+1)} = y_j^{(v)} + u_j^{(v+1)}. \tag{7}$$

Here, u_j and y_j denote particle velocity and position, q_j^{best} is the best position found by particle j, g^{best} is the global best position, ω is the inertia weight, d_1 and d_2 are acceleration parameters, and s_1 and s_2 are random values.

The architecture can be extended with additional inputs for particular use cases. Its components transform input factors into a cash-demand estimate through hidden layers and optimization. Input variables are normalized to a common range using

$$y'_j = \frac{y_j - \min(y)}{\max(y) - \min(y)}. \tag{8}$$

The output of the ith hidden neuron can be expressed as

$$w_i = \sigma\left(\sum_{j=1}^{m} z_{i,j} y_j + c_i\right). \tag{9}$$

The sigmoid function used for this transformation is

$$\sigma(y) = \frac{1}{1+e^{-y}}. \tag{10}$$

Alternatively, a hyperbolic tangent activation may be used:

$$\sigma(y) = \frac{e^{y} - e^{-y}}{e^{y} + e^{-y}}. \tag{11}$$

The output layer aggregates hidden-neuron results to form the forecast:

$$x = \sum_{i=1}^{n} z_i^{\text{out}} w_i + c^{\text{out}}. \tag{12}$$

The model's loss is the mean squared error over M observations:

$$K = \frac{1}{M} \sum_{l=1}^{M} \left(x_l - x_{\text{true},l}\right)^2. \tag{13}$$

The actual cash demand can also be normalized as

$$x'_{\text{true}} = \frac{x_{\text{true}} - \min(x_{\text{true}})}{\max(x_{\text{true}}) - \min(x_{\text{true}})}. \tag{14}$$

For backpropagation, the net input to the ith hidden neuron is

$$v_i = \sum_{j=1}^{m} z_{i,j} y_j + c_i. \tag{15}$$

The sigmoid derivative is

$$\sigma'(y) = \sigma(y)\left(1 - \sigma(y)\right), \tag{16}$$

and the derivative of the hyperbolic tangent activation is

$$\sigma'(y) = 1 - \tanh^2(y). \tag{17}$$

These gradients are used to adjust weights and biases in the hidden and output layers. The study also gives the regularized loss as

$$K_{\text{reg}} = L + 2\sum_{m,n} z_{m,n}^2. \tag{18}$$

ANN-CFM incorporates holiday effects, weekdays of the month, salary-day effects, working-day offsets, month of year, and prior-year data. After normalization, these inputs pass through nonlinear activation functions in the hidden layer and are aggregated in the output layer. Backpropagation updates weights and biases to correct inaccurate predictions, while PSO supports faster convergence and parameter optimization. The combined design is intended for large, uncertain, and changing cash-flow environments.

6. RESULTS

The ANN-CFM model is evaluated across cash-flow conditions that vary with social and financial factors, seasonal changes, and regional events. Its ability to learn from data and update its parameters is presented as a way to address limitations of linear assumptions. The following tables and figures reproduce the comparative values reported in the study.

Table 2. Model parameter valuation with ANN-CFM

Working days	Normal	PSO-CMM	ANN-CFM
1	27	25	14
2	21	19	13
3	17	16	12
4	13	16	14
5	12	14	13
6	14	13	11
7	10	9	8
8	11	10	8
9	8	9	8

Table 2 shows that ANN-CFM values are generally lower than those of Normal and PSO-CMM across the working days.

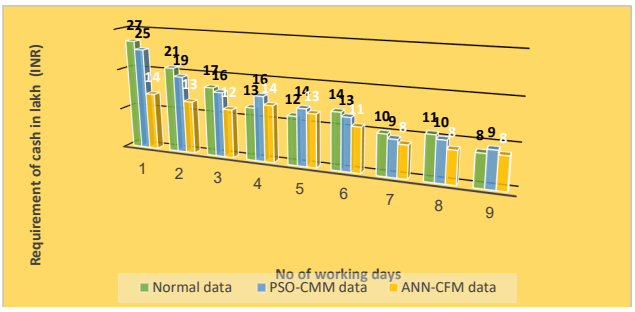


Figure 4. Forecasting of short-term financial flows.

Table 3. ANN-CFM model parameters across working days

Working days	Normal	PSO-CMM	ANN-CFM
10	51	49	13
20	23	22	12
30	21	20	11
40	53	50	10
50	24	23	10
6	22	21	9
70	56	54	9
80	25	24	9
90	23	22	9

The second comparison reports Normal values ranging from 21 to 56, PSO-CMM values from 20 to 54, and ANN-CFM values from 9 to 13. The study interprets the ANN-CFM series as the most optimized of the three approaches for the reported cases.

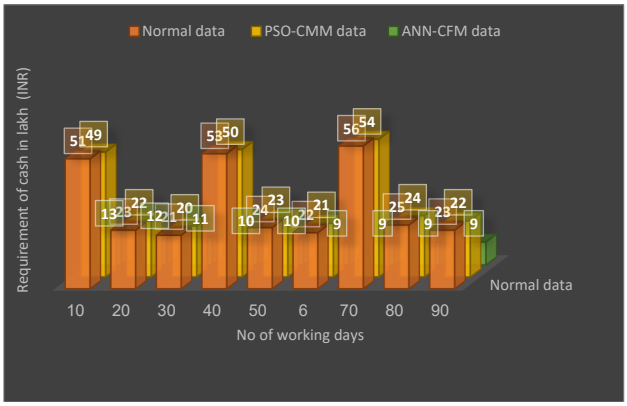


Figure 5. Forecasting of longer-term financial flows.

Table 4. Comparison of default and ANN-CFM variables

Neurons	Default	ANN-CFM
13	0.0222	0.0102
12	0.0227	0.0100
11	0.0182	0.0080
10	0.0189	0.0086
9	0.0197	0.0087

Table 4 and Figure 6 compare default variable values with ANN-CFM values across five neuron counts. In every reported case, the ANN-CFM value is lower. For example, at 13 neurons the value decreases from 0.0222 to 0.0102, and at 11 neurons it decreases from 0.0182 to 0.0080. The paper relates these reductions to improved parameter adjustment and computational efficiency.

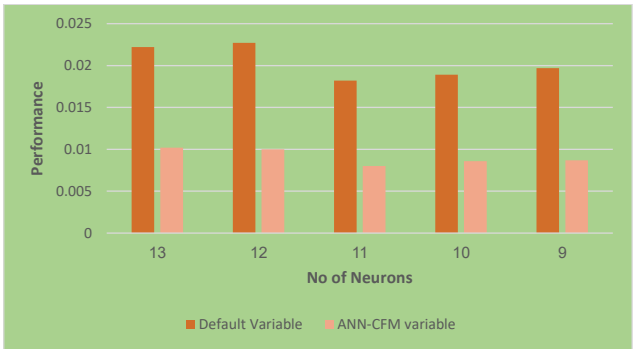


Figure 6. Efficacy of different systems.

Table 5. Evaluation criteria for ANN-CFM

Forecast model	MAE	MAPE
Long-term	2.7799	16.4534
Short-term	2.2355	7.3434

The short-term forecast has lower reported error than the long-term forecast: its MAE is 2.2355 and MAPE is 7.3434, compared with 2.7799 and 16.4534, respectively, for the long-term forecast. The study attributes the difference to the greater complexity and uncertainty of longer prediction horizons and concludes that ANN-CFM can adapt to forecasts over different periods.

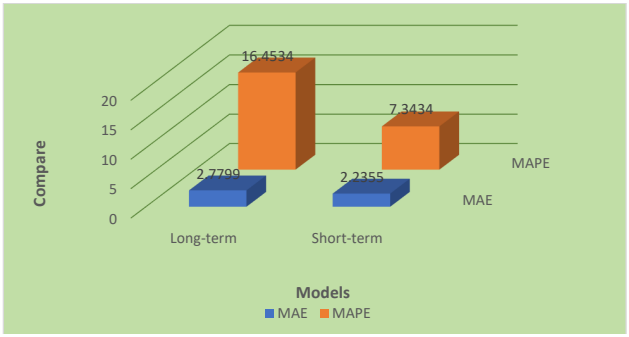


Figure 7. Effectiveness of different models.

7. CONCLUSION

ANN-CFM is presented as a method for addressing cash-management challenges in financial organizations. It uses neural networks to learn linear and nonlinear patterns in cash demand and incorporates salary, holiday, and working-day effects. Backpropagation and PSO are used to optimize parameters and reduce forecast errors, with the objective of limiting risks associated with insufficient or excessive cash balances. The flexible architecture is intended to adapt to different financial requirements and forecasting horizons.

The reported results indicate improved accuracy and flexibility compared with the approaches considered in the study. Further work is proposed to include additional exogenous factors, such as economic indicators, environmental conditions, and neighbourhood trends. Distributed or cloud-computing platforms are also suggested to support calculations on larger datasets.

REFERENCES

[1] M. Qu and Y. Li, "Financial risk early-warning model based on kernel principal component analysis in public hospitals," *Mathematical Problems in Engineering*, vol. 2021, no. 1, pp. 1–7, 2021.

[2] Q. Wang, F. Hui, X. Wang, and Q. Ding, "Research on early warning and monitoring algorithm of financial crisis based on fuzzy cognitive map," *Cluster Computing*, vol. 22, no. S2, pp. 3689–3697, 2019.

[3] H. Zeng, "Influences of mobile edge computing-based service preloading on the early-warning of financial risks," *The Journal of Supercomputing*, vol. 78, no. 9, pp. 11 621–11 639, 2022.

[4] Z. Xu, G. Zhu, N. Metawa, and Q. Zhou, "Machine learning based customer meta-combination brand equity analysis for marketing behavior evaluation," *Information Processing & Management*, vol. 59, no. 1, 2022.

[5] X. Wang, X. Yu, L. Guo, F. Liu, and L. Xu, "Student performance prediction with short-term sequential campus behaviors," *Information*, vol. 11, no. 4, p. 101, 2020.

[6] Q. Guo, Z. Zhu, Q. Lu, D. Zhang, and W. Wu, "A dynamic emotional session generation model based on seq2seq and a dictionary-based attention mechanism," *Applied Sciences*, vol. 10, no. 6, pp. 1–10, 2020.

[7] H. Ren, X. Mao, W. Ma, J. Wang, and L. Wang, "An english-chinese machine translation and evaluation method for geographical names," *ISPRS International Journal of Geo-Information*, vol. 9, no. 3, pp. 193–201, 2020.

[8] X. Nie and G. Deng, "Enterprise financial early warning based on lasso regression screening variables," *Journal of Financial Risk Management*, 2020.

[9] P. K. Jadwal, S. Jain, U. Gupta, and P. Khanna, "K-means clustering with neural networks for atm cash repository prediction," in *International Conference on Information and Communication Technology for Intelligent Systems*, 2017, pp. 588–596.

[10] K. Venkatesh, V. Ravi, A. Prinzie, and D. Van den Poel, "Cash demand forecasting in atms by clustering and neural networks," *European Journal of Operational Research*, vol. 232, no. 2, pp. 383–392, 2014.

[11] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[12] F. Kiaee, C. Gagné, and M. Abbasi, "Alternating direction method of multipliers for sparse convolutional neural networks," *arXiv preprint arXiv:1611.01590*, 2016.

[13] F. Kiaee, H. Fahimi, and H. Rabbani, "Intra-retinal layer segmentation of optical coherence tomography using 3d fully convolutional networks," in *2018 25th IEEE International Conference on Image Processing*, 2018, pp. 2795–2799.

[14] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," *arXiv preprint arXiv:1509.06461*, 2015.

[15] F. Kiaee, "Integration of electric vehicles in smart grid using deep reinforcement learning," in *2020 11th International Conference on Information and Knowledge Technology*, 2020, pp. 40–44.

[16] K. L. López, C. Gagné, and M.-A. Gardner, "Demand-side management using deep learning for smart charging of electric vehicles," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2683–2691, 2018.

[17] O. B. Sezer, A. M. Ozbayoglu, and E. Dogdu, "An artificial neural network-based stock trading system using technical analysis and big data framework," in *Proceedings of the South-East Conference (ACM SE '17)*, 2017, pp. 223–226.

[18] G. I. Sher, "Forex trading using geometry sensitive neural networks," in *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation (GECCO '12)*, 2012, pp. 1533–1534.

[19] W. Farsal, S. Anter, and M. Ramdani, "Deep learning: An overview," in *Proceedings of the 12th International Conference on Intelligent Systems: Theories and Applications (SITA '18)*, 2018, pp. 1–6.

[20] A. Ng, "Deep learning: What's next," in *Proceedings of the 2016 International Conference on Autonomous Agents and Multiagent Systems*, 2016.

[21] R. Salakhutdinov, "Deep learning," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, p. 1973.

[22] J. Sang, "Deep learning interpretation," in *Proceedings of the 26th ACM International Conference on Multimedia*, 2018, pp. 2098–2100.

[23] Y.-H. Chou, S.-Y. Kuo, and Y.-T. Lo, "Portfolio optimization based on funds standardization and genetic algorithm," *IEEE Access*, vol. 5, pp. 21 885–21 900, 2017.

[24] A. M. Rather, V. N. Sastry, and A. Agarwal, "Portfolio selection using maximum-entropy gain loss spread model: A ga based approach," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2014, pp. 400–406.

[25] F. Z. Habbab, M. Kampouridis, and A. A. Voudouris, "Optimizing mixed-asset portfolios involving reits," in *Proceedings of the IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr)*, 2022, pp. 1–8.