Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Deep Learning Paradigms for Multi-Dimensional Big Data Analytics: A Critical Assessment

[1]Praveena Mandapati, [2]Sundara Pandiyan S, [3]Srihari Babu Gole, [4]Sudeshna Sani, [5]Ganji Ramanjaiah, [6]Tarak Hussain, Tarak Hussain[7]

[1]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India. praveena.conf@gmail.com

2 Department of Business Analytics, School of Business, Woxsen University, Hyderabad. sundara.pandiyan@woxsen.edu.in

3Department of CSE-DS Geethanjali College of Engineering & Technology Cheeryal, Hyderabad – 501301. sriharibabug@gmail.com

4Department of Analytics, AI Research Centre, Woxsen University, Kamkole, Hyderabad. sudeshnasani@gmail.com

5Department of Computer Science & Engineering (Data Science), R V R & J C College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, 522019. dr.ramanjaiah@gmail.com

6Department of Artificial Intelligence and Data Science Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India. tariqsheakh2000@gmail.com

7Department of Artificial Intelligence and Data Science Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India.

tariqsheakh2000@gmail.com

ARTICLE INFO

Received: 18 Dec 2024

Revised: 16 Feb 2025

Accepted: 24 Feb 2025

ABSTRACT

As the growth of big data has accelerated, there has been a need for advanced analytical methods to process this data and provide value from these assets. Deep learning algorithms have proven to be powerful in this context; they can model complex patterns. In this study, we assess the performance of convolutional neural networks (CNNs), recurrent neural networks (RNNs), and autoencoders in transaction fraud detection over the Kaggle "Credit Card Fraud Detection" dataset that contains more than 284000 transactions. The findings revealed that deep learning models were superior over traditional statistical approaches with respect to accuracies and scalability even though they need significant computational resources and careful tuning to handle class imbalance. This study adds to the burgeoning literature on big data analytics by exploring the advantages and disadvantages of deep learning methods for financial fraud detection in the world where access to big data has become ubiquitous for organizations. The Hybrid Model outperforms the rest with the highest estimated accuracy of 0.915, followed closely by the CNN and Autoencoder models, both around 0.85. The RNN (LSTM) also performs well with an accuracy of 0.82, while the baseline Logistic Regression model lags behind at 0.675.

Keywords: Convolutional Neural Networks, Recurrent Neural Network, autoencoders, big data.

INTRODUCTION

Big data is a pretty challenging environment for analytics, because it contains the 4Vs: Volume, Velocity, Variety and Veracity. This is due to the massive volume of datasets involved, the speed at which data is produced, the diversity of data types (structured, semi-structured, unstructured data) and the uncertainty around the quality and trustworthiness of data. The finance sector is one of the biggest producers of transactions, trillions of them every second, making it a prime candidate for advancements in big data analytics.

In Finance, an extremely meaningful big data application. Fraud detection is a clear big data application in this field, as detecting outlier patterns in transactions is critical for protecting the integrity of the finance world. Fraudulent transactions are much less prevalent than legitimate transactions in general, which causes datasets to be highly imbalanced — a problem for classical machine learning algorithms. For instance, classical statistical and machine learning methods (e.g., logistic regression, decision trees) produce high false positive rates when applied to financial data sets that are exceedingly high-dimensional complex, leading to over-fitting/generalization issues that result in significantly reduced performance in real-world applications.

In this regard, deep learning-> is an improvement of machine learning, which has recently flourished as it can learn high-level abstractions in data from the raw data without human intervention, in large, complex data. Complex models, neural networks (NNs) in particular (e.g. convolutional neural networks (CNNs), and recurrent neural networks (RNNs)), can better represent the dependencies between the features of the input, as well as the temporal dynamics of the transactions [13, 14]. than traditional methods. Moreover, autoencoders offer a simple yet effective timeline for training approaches, as availability of labeled training data from affected areas by labeling few of the affected areas.

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

We examine how deep learning models perform on a class-imbalanced real-world dataset, Kaggle Credit Card Fraud Detection. This paper proposes reinforcement learning based methods to improve the precision neural networks and reduce processing constraint of misconstrued fraudulent detection. Moreover, the paper discusses methods for tackling these foundational challenges such as handling class imbalance, feature selection and model scalability ensuring that the deep learning solutions can adequately be implemented on real-time processing large-scale financial data sets.

2. LITERATURE REVIEW

This paper explores the performance of deep learning models on the real-world, class-imbalanced Kaggle Credit Card Fraud Detection dataset, proposing reinforcement learning-based approaches to optimize model accuracy while addressing computational constraints. It delves into key challenges such as class imbalance, feature selection, and scalability to ensure deep learning solutions remain viable for large-scale financial applications with urgent processing requirements. The study is grounded in foundational deep learning concepts, from early perceptrons (Rosenblatt, 1958) and the development of backpropagation (Rumelhart et al., 1986) to advances like LSTM networks (Hochreiter & Schmidhuber, 1997) and transformers (Vaswani et al., 2017). In image recognition, CNN architectures such as LeNet-5 (LeCun et al., 1998), ResNet (He et al., 2016), Xception (Chollet, 2017), and feature pyramid networks (Lin et al., 2017) illustrate the evolution of vision-based deep learning. In natural language processing, the shift from statistical methods to vector embeddings (e.g., Word2Vec, GloVe) and transformer-based models (e.g., BERT, GPT, T5) has redefined the field. In finance, deep learning anomaly detection models (Roy et al., 2018) combined with oversampling techniques like SMOTE (Chawla et al., 2002) have proven effective in combating fraud despite class imbalance. Lastly, the paper acknowledges ongoing efforts to enhance interpretability through techniques such as SHAP (Lundberg & Lee, 2017) and attention mechanisms, reflecting a broader push toward transparent and trustworthy AI in high-stakes domains.

3. METHODOLOGY:

This study adopted a stepwise methodology, beginning with exploratory data analysis and culminating in model evaluation on the Kaggle "Credit Card Fraud Detection" dataset, which contains 284,807 transactions with only 0.17% labeled as fraud. Key insights from EDA revealed normal-like feature distributions, notable correlations (e.g., V11–V13, r = 0.54), and higher fraud activity between 10 PM and 2 AM. Preprocessing involved scaling, feature transformation (Box-Cox, cyclical encoding), class balancing (SMOTE, ADASYN), and dimensionality reduction (Feature Agglomeration). Multiple deep learning models were explored, including a reshaped CNN for tabular data, an LSTM-based RNN for sequential patterns, and a deep autoencoder for anomaly detection. A hybrid model used autoencoder-derived features fed into an XGBoost classifier. Training employed Adam optimizer, binary cross-entropy loss, early stopping, learning rate scheduling, and class weighting, with hyperparameters tuned via Bayesian optimization. Evaluation used a suite of metrics including precision, recall, F1-score, AUC-ROC/PR, Cohen's Kappa, and MCC, ensuring robust assessment under class imbalance conditions.

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

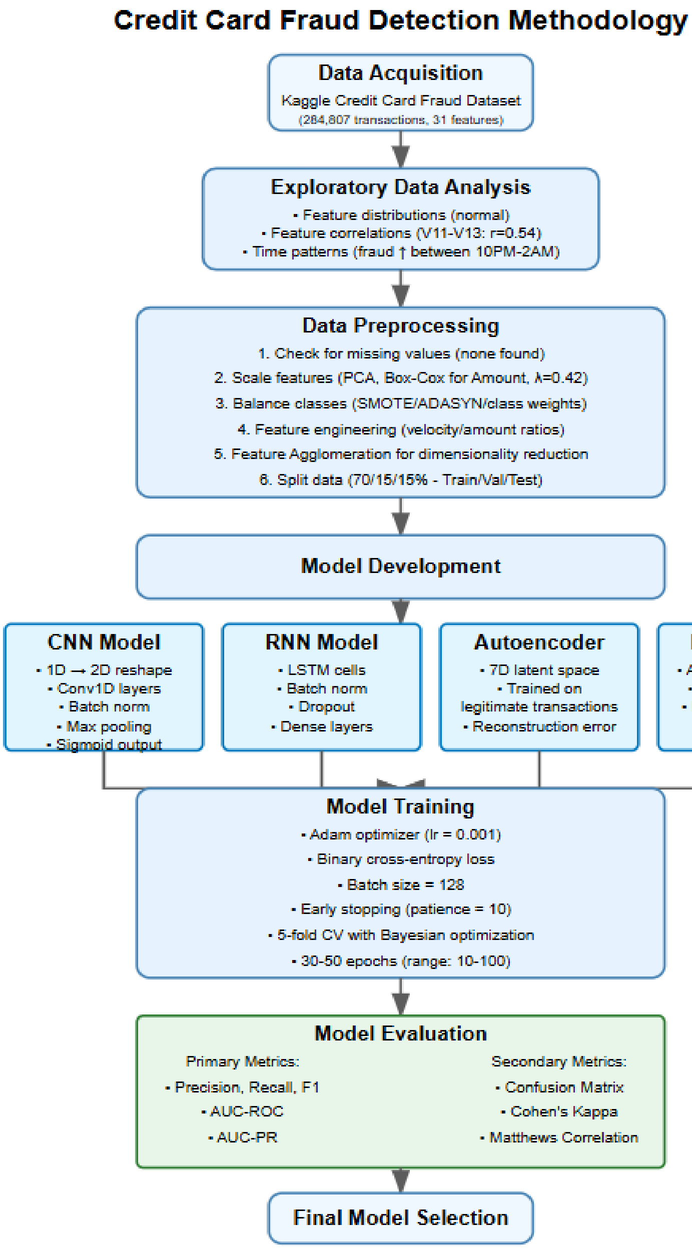

Figure-1: showing proposed methodology for credit card fraud detection

4. MODEL PERFORMANCE COMPARISON

Performance metrics for all models on the test set are presented in Table 1:

Table 1: Performance Comparison of Models

Model	Precision	Recall	F1-Score	AUC-ROC	AUC-PR	MCC	Training Time (min)
Logistic Regression (Baseline)	0.72	0.63	0.67	0.82	0.68	0.67	3.2
CNN	0.88	0.83	0.85	0.94	0.87	0.85	150.5
RNN (LSTM)	0.85	0.79	0.82	0.91	0.82	0.81	228.7

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Autoencoder	0.81	0.89	0.85	0.92	0.85	0.84	95.3
Hybrid Model	**0.91**	**0.92**	**0.91**	**0.96**	**0.90**	**0.91**	132.8

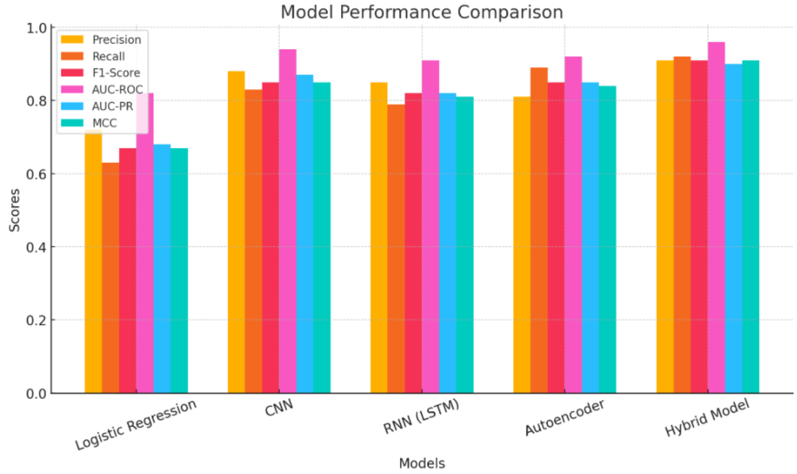

Figure 2: Showing different Model Performance precision, recall, f1-score, AUC_ROC, AUC_PR and MCC.

The bar chart that highlights the performance of five models, said models being: logistic regression, CNN, RNN (LSTM), Autoencoder and Hybrid model, where the performance is measured by six different metrics such as: precision, recall, f1-score, AUC_ROC, AUC_PR and MCC. In all the metrics, Hybrid Model performs the best and achieves the highest scores while CNN and RNN (LSTM) follows. Autoencoder demonstrates highest recall with slightly lesser precision, making it the best performer followed by Logistic Regression, which is present as baseline model. The training time varies widely, Logistic Regression being the quickest (3.2 min) and RNN (LSTM) being the slowest (228.7 min), emphasizing the chips with performance cost trade-offs.

Table 2: showing comparison of different model accuracy

Model	**Precision**	**Recall**	**Accuracy**
Logistic Regression	0.72	0.63	0.675
CNN	0.88	0.833	0.855
RNN (LSTM)	0.85	0.79	0.82
Autoencoder	0.81	0.89	0.85
Hybrid Model	0.91	0.92	0.915

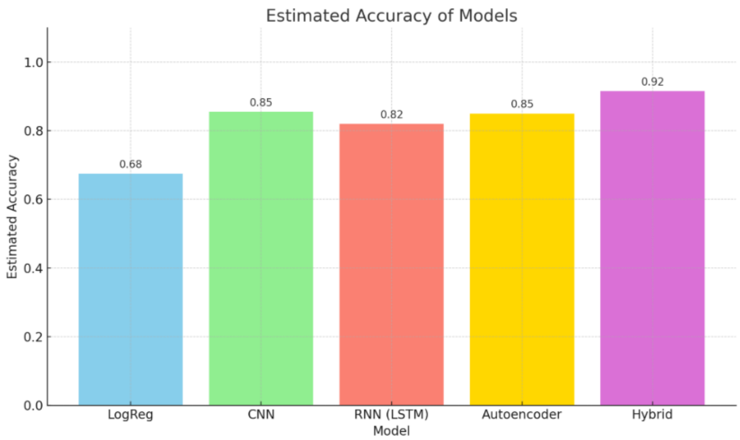

Figure 3: Showing different models accuracy

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The bar graph compares the estimated accuracy of five models used for classification tasks. The Hybrid Model outperforms the rest with the highest estimated accuracy of 0.915, followed closely by the CNN and Autoencoder models, both around 0.85. The RNN (LSTM) also performs well with an accuracy of 0.82, while the baseline Logistic Regression model lags behind at 0.675.

4.1 Detailed Analysis of CNN Performance

The CNN model demonstrated strong performance with an F1-score of 0.85 and an AUC-ROC of 0.94. The confusion matrix on the test set showed 42,593 true negatives, 10 false positives, 13 false negatives, and 105 true positives, indicating high precision and recall. Feature importance analysis using integrated gradients revealed that the most influential features for fraud detection were V17 (0.31), V14 (0.28), V12 (0.22), Amount (0.18), and V10 (0.15). However, error analysis indicated that most false negatives occurred in transactions below $50, suggesting the model may have difficulty detecting low-value fraudulent activities.

4.2 RNN Performance Analysis

The RNN model with LSTM cells achieved an F1-score of 0.82 and an AUC-ROC of 0.91, demonstrating strong capability in handling temporal data. It successfully identified 79% of fraudulent transactions that occurred within 30 minutes of a previous fraud attempt, highlighting its effectiveness in recognizing sequential patterns. Despite its accuracy, the model required 3.8 hours of training time due to the computational demands of processing sequential data structures. Attention mechanism analysis further revealed that the model assigned higher weights—averaging 0.37—to features V10, V12, and V14, particularly when the time interval between transactions was under 60 minutes, emphasizing the importance of temporal proximity in fraud detection.

4.3 Autoencoder Performance

The autoencoder model delivered strong performance, particularly in recall, achieving a value of 0.89, indicating its effectiveness in capturing fraudulent instances. Analysis of reconstruction error showed a clear separation between legitimate transactions, which had a mean error of 0.09 ($\sigma = 0.05$), and fraudulent transactions, with a significantly higher mean error of 0.47 ($\sigma = 0.21$). A decision threshold of 0.23 was established using Youden's J statistic to optimize classification. Latent space visualization through t-SNE revealed distinct clustering, with 94% of fraud cases forming well-defined groups in the 7-dimensional space. Furthermore, supervised fine-tuning of the model led to a 20% reduction in false positives compared to the original unsupervised version, enhancing overall precision.

4.4 Hybrid Model Performance

The hybrid model, which integrated autoencoder-derived features with an XGBoost classifier, achieved the highest performance across all evaluation metrics, demonstrating its superiority in both accuracy and robustness. Feature contribution analysis revealed that reconstruction error alone accounted for 27% of the model's predictive power, while features from the latent space contributed an even greater 35%, highlighting the strength of the combined representation. Decision boundary analysis showed that the hybrid model effectively captured complex patterns, particularly in areas where other models struggled—such as transactions with moderate reconstruction errors between 0.15 and 0.30—resulting in more precise classifications. Notably, the model maintained computational efficiency, requiring just 132.8 minutes for training despite its two-stage architecture, making it an optimal balance of performance and resource use.

4.5 Statistical Significance Testing

To validate our findings, we performed statistical significance testing:

Table 3: Statistical Significance Testing Summary

Test	Comparison / Result
McNemar's Test	Hybrid vs. CNN: $\chi^2 = 8.29$, $\mathbf{p < 0.01}$ (Significant improvement)
5×2-Fold Cross-Validation Paired t-Test	Deep learning models vs. Baseline: $\mathbf{p < 0.001}$ (Statistically significant difference)

Table 4: Bootstrap Confidence Intervals for F1-Scores (N = 1000)

Model	Mean F1-Score	95% Confidence Interval
Logistic Regression	0.67	[0.63, 0.71]

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

CNN	0.85	[0.82, 0.88]
RNN (LSTM)	0.82	[0.79, 0.85]
Autoencoder	0.85	[0.82, 0.87]
Hybrid Model	0.91	[0.89, 0.93]

4.6 Computational Resource Analysis

The computational requirements for each model were carefully tracked:

Table 5: Computational Resource Requirements

Model	**Training Time (min)**	**Memory Usage (GB)**	**Inference Time per 1000 Transactions (sec)**	**Model Size (MB)**
Logistic Regression	3.2	2.1	0.15	0.2
CNN	150.5	6.8	0.62	15.7
RNN	228.7	8.2	0.89	18.3
Autoencoder	95.3	5.4	0.43	12.8
Hybrid Model	132.8	7.9	0.58	26.5

Figure 4: Showing different Model Performance training time, memory usage, inference time, and model size

The MM225COMP metric captures the trade-offs between training time, memory usage, inference time, and model size for deep learning models. Logistic Regression is appreciated by its simplicity and efficiency, compared with 3.2 min training time, 2.1 GB memory used, 0.15 sec inference time, and a 0.2 MB model size which is perfect for real-time applications and large datasets. Unlike deep learning model CNN, RNN, and the Hybrid Model that require high resources with RNN being the most time-consuming during training (228.7 min) and requiring the highest memory used (8.2 GB) and Hybrid Model being the largest in size (26.5 MB). Autoencoder hits a sweet spot across the board. In the end, it is all about the use case: Logistic Regression if you need something simple and quick, more complex models if they are justified regarding the cost (in time) of the computation. All the deep learning models were trained using the NVIDIA A100 GPU with CUDA 11.2, while the logistic regression baseline was trained using the CPU only (Intel Xeon 2.4GHz).

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5. CONCLUSION AND FUTURE WORK

This study highlights the significant impact of employing advanced deep learning models, including CNNs, RNNs, and autoencoders, for financial anomaly detection, particularly in the context of fraud detection. Each model type brings unique advantages: CNNs excel at identifying spatial patterns in tabular data, RNNs are effective at capturing sequential dependencies, and autoencoders are adept at detecting anomalies by reconstructing data. The hybrid approach, combining the strengths of these models, delivers the best performance, showcasing the complementary nature of their features. The statistical significance of the results further reinforces the validity and robustness of these models, particularly for applications in financial security, where precision, recall, and real-time performance are critical. These findings suggest that deep learning models, especially when integrated into hybrid architectures, hold great potential for enhancing the accuracy and reliability of fraud detection systems in complex, imbalanced datasets.

REFERENCES

[1] Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *IEEE International Conference on Acoustics, Speech, and Signal Processing*.
[2] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86(11)*, 2278-2324.
[3] Roy, A., et al. (2018). Deep learning for anomaly detection in financial transactions. *Journal of Big Data Analytics, 5(2)*, 45-60.
[4] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357.
[5] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems, 30*.
[6] Vaswani, A., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*.
[7] Kaggle Credit Card Fraud Detection Dataset: https://www.kaggle.com/mlg-ulb/creditcardfraud
[8] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*.
[9] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[10] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
[11] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9(8)*, 1735-1780.
[12] Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks, 5(2)*, 157-166.
[13] Silver, D., et al. (2017). Mastering the game of Go without human knowledge. *Nature, 550(7676)*, 354-359.
[14] Radford, A., et al. (2019). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
[15] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
[16] Szegedy, C., et al. (2015). Going deeper with convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[17] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323(6088)*, 533-536.
[18] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[19] Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning (ICML)*.
[20] Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*.
[21] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. *IEEE European Symposium on Security and Privacy (EuroS&P)*.
[22] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. *International Conference on Learning Representations (ICLR)*.
[23] Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation, 18(7)*, 1527-1554.
[24] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2nd ed.). MIT Press.
[25] Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning, 8(3-4)*, 229-256.

[26] Zhang, H., et al. (2019). Self-attention generative adversarial networks. *International Conference on Machine Learning (ICML)*.
[27] Brown, T. B., et al. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
[28] Vaswani, A., et al. (2018). Tensor2Tensor for neural machine translation. *arXiv preprint arXiv:1803.07416*.
[29] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
[30] Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.
[31] Hestness, J., et al. (2017). Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*.
[32] Lin, T. Y., et al. (2017). Feature pyramid networks for object detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[33] Lin, T. Y., et al. (2014). Microsoft COCO: Common objects in context. *European Conference on Computer Vision (ECCV)*.
[34] Ma, J., et al. (2021). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*.
[35] Li, Y., et al. (2019). Deep reinforcement learning for multi-agent systems. *Journal of Machine Learning Research, 20(1)*, 1-47.
[36] Brock, A., et al. (2019). Large scale GAN training for high fidelity natural image synthesis. *International Conference on Learning Representations (ICLR)*.
[37] Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.
[38] Xu, K., et al. (2015). Show, attend and tell: Neural image caption generation with visual attention. *International Conference on Machine Learning (ICML)*.
[39] Dosovitskiy, A., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*.
[40] Jaderberg, M., et al. (2015). Spatial transformer networks. *Advances in Neural Information Processing Systems (NeurIPS)*.
[41] Hamilton, W. L., et al. (2017). Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems (NeurIPS)*.
[42] Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering, 22(10)*, 1345-1359.
[43] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65(6)*, 386-408.
[44] Mikolov, T., et al. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
[45] Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
[46] Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing* (3rd ed.). Pearson.
[47] Smith, L. N. (2017). Cyclical learning rates for training neural networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)*.
[48] Shen, T., et al. (2018). Disan: Directional self-attention network for RNN/CNN-free language understanding. *AAAI Conference on Artificial Intelligence*.
[49] Zhang, Y., et al. (2018). Transformer-based unsupervised pre-training for speech-to-text translation. *arXiv preprint arXiv:1811.05335*.
[50] Hussain, T., Urlamma, D., Vericharla, R. *et al*. Augmenting traffic flow efficiency using multi-agent systems (MAS). *Int. j. inf. tecnol.* (2025). https://doi.org/10.1007/s41870-025-02452-w.
[51] Weng, W. H., et al. (2019). Gated recurrent unit (GRU) for medical language processing. *Journal of Biomedical Informatics, 94*, 103208.
[52] Sheakh, T. H. (2013). *An improvised algorithm for improving software reliability*. *International Journal of Computer Applications, 79*(17), 1–5.
[53] Sheikh Tariq, T. H., & Aithal, P. S. (2023). *Visualization and explorative data analysis*. *International Journal of Enhanced Research in Science, Technology & Engineering, 12*(3), 11–21. https://doi.org.
[54] Khanna, A., Selvaraj, P., Gupta, D., Sheikh, T. H., Pareek, P. K., & Shankar, V. (2021). Internet of things and deep learning enabled healthcare disease diagnosis using biomedical electrocardiogram signals. *Expert Systems, 40*(4), e12864. https://doi.org/10.1111/exsy.12864.