American Journal of Technology Advancement Vol.2, No.6 (June, 2025),
E-ISSN: 2997-9382
American Journal of Technology Advancement
https://semantjournals.org/index.php/AJTA

| Research Article

Check for updates

Dynamic Object Identification using Neural Networks

Sabirov Ulugbek Kuchkarovich
Associate Professor, Andijan State Technical Institute
uqsabirovasu1951@gmail.com

Annotation
Dynamic object identification plays a critical role in a wide range of engineering applications, including autonomous navigation, robotics, and adaptive control systems. Traditional identification techniques often struggle when dealing with nonlinearities, time delays, and system uncertainties inherent in dynamic environments. Recently, neural networks have emerged as a promising solution due to their ability to approximate complex nonlinear mappings and learn from noisy data in real time. This study explores the use of different neural network architectures—specifically, Feedforward Neural Networks (FNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks—for the identification of dynamic objects whose parameters change over time. Simulation experiments using both synthetic and real-world datasets demonstrate that LSTM networks provide superior accuracy and robustness, particularly in capturing temporal dependencies and sudden transitions in system behavior. The findings suggest that neural networks can serve as effective alternatives to traditional methods, enabling adaptive modeling and prediction in complex dynamic environments. This paper also discusses the implications of network architecture choice, training data quality, and hyperparameter tuning on overall system performance.

Keywords: Dynamic objects, neural networks, identification, nonlinear systems, real-time processing, recurrent networks, control systems.

Introduction. The identification of dynamic systems and objects is a fundamental problem in control theory, signal processing, and artificial intelligence. In many real-world applications, such as autonomous vehicles, robotics, aerospace navigation, and industrial automation, the system to be controlled or observed is subject to continuous changes over time. These changes can result from varying environmental conditions, internal dynamics, or external disturbances, making the modeling and prediction of such systems particularly challenging.

Traditional model-based identification techniques, including least-squares estimators, state observers, and Kalman filters, rely on accurate mathematical representations of system dynamics. While effective under certain conditions, these methods tend to perform poorly in nonlinear, high-dimensional, or noisy environments. Moreover, when the structure of the system is not fully known or is time-varying, traditional approaches require frequent re-calibration or manual intervention, which limits their adaptability and scalability [1].

Neural networks, on the other hand, have demonstrated significant promise as data-driven alternatives for system identification. Their ability to approximate arbitrary nonlinear functions, combined with powerful learning algorithms and parallel processing capabilities, make them particularly suited for modeling dynamic systems. Among the various neural architectures, Recurrent Neural Networks (RNNs) and their advanced variant, Long Short-Term Memory (LSTM) networks, are well-known for their capability to capture temporal dependencies in sequential data. This makes them ideal candidates for tracking the evolving behavior of dynamic objects over time [2].

This paper investigates the use of neural networks for the identification of dynamic objects whose characteristics change continuously or discretely. We explore how different network architectures perform under varying conditions and analyze their ability to adapt, generalize, and predict future system states. By comparing these methods through simulations and empirical tests, we aim to provide insights into the practical deployment of neural networks in adaptive control and real-time identification systems [3].

Methodology. In order to evaluate the effectiveness of neural networks for identifying dynamic objects, a comprehensive methodological framework was developed. The approach involves multiple stages including data acquisition, preprocessing, model design, training, evaluation, and validation. Each step is carefully tailored to ensure the accurate representation of dynamic behavior and to assess the generalization capabilities of the neural networks.

Synthetic datasets were generated using MATLAB Simulink models that simulate dynamic systems with varying characteristics, such as time-varying mass, damping coefficients, and nonlinear external forces. In addition, real-world datasets were collected from sensors tracking the position and velocity of mobile objects under dynamic conditions. Each dataset consisted of multivariate time-series inputs and outputs [4].

Prior to model training, all data were normalized within the range [0, 1] to facilitate faster convergence during the learning process. The datasets were partitioned into training (70%), validation (15%), and test (15%) sets to enable unbiased model evaluation.

Three types of neural network architectures were implemented:

- Feedforward Neural Network (FNN): A basic multilayer perceptron (MLP) with one hidden layer and ReLU activation functions.
- Recurrent Neural Network (RNN): Designed to capture temporal dependencies in the input sequences using simple recurrent units.
- Long Short-Term Memory (LSTM): A specialized type of RNN that uses memory cells to learn long-term dependencies and mitigate the vanishing gradient problem.

All networks were implemented in Python using TensorFlow and Keras libraries [5].

Each model was trained using the Adam optimizer with a learning rate of 0.001 and mean squared error (MSE) as the loss function. The number of epochs ranged from 50 to 200 depending on the network architecture, and early stopping was used to avoid overfitting. Batch sizes of 32 and 64 were tested for performance optimization.

Hyperparameter tuning was conducted through grid search, including adjustments in the number of hidden units, number of layers, dropout rates, and activation functions. Regularization techniques such as L2 weight decay and dropout were employed to improve model robustness.

The models were evaluated using the following metrics:

- Root Mean Square Error (RMSE) – to assess prediction accuracy.
- Mean Absolute Error (MAE) – to measure average deviation.
- R^2 Score – to evaluate the proportion of variance explained by the model.
- Computation Time – to assess real-time feasibility.

Cross-validation was performed using a k-fold approach (k=5) to ensure statistical reliability. Additionally, sensitivity analysis was conducted to test how changes in input dynamics affected model performance [6].

Results and Discussion. The experiments conducted in this study yielded significant insights into the performance of different neural network architectures for dynamic object identification. The results are presented in terms of prediction accuracy, computational efficiency, and generalization capability under different dynamic conditions.

Among the three neural network models tested—Feedforward Neural Network (FNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM)—the LSTM-based model achieved the highest prediction accuracy. For dynamic systems characterized by rapid fluctuations or nonlinear transitions, the LSTM model consistently outperformed both FNN and simple RNN. The average Root Mean Square Error (RMSE) for LSTM across multiple datasets was approximately 0.028, compared to 0.065 for RNN and 0.089 for FNN. Similarly, the Mean Absolute Error (MAE) was lowest in LSTM models, indicating better tracking of real-time object behavior [7].

The improved accuracy of LSTM can be attributed to its memory cell structure, which allows the model to retain long-term temporal dependencies. In contrast, FNNs—lacking any memory capability—were only effective in modeling static or slowly varying dynamics. RNNs, while able to handle temporal data, suffered from vanishing gradient issues during long sequences, limiting their effectiveness.

Generalization tests were conducted by introducing previously unseen input patterns and varying noise levels. LSTM models showed strong adaptability and continued to provide reliable predictions even when the input sequences exhibited significant variability. This is crucial in real-world applications where sensor measurements are often noisy and inconsistent [8].

Furthermore, models trained on one dynamic system were tested on structurally similar but behaviorally different systems. LSTM networks demonstrated superior transfer learning capabilities, requiring fewer retraining epochs to adapt to new system dynamics. This reinforces their practicality in environments where system parameters may shift over time.

While LSTM networks provided the best performance in terms of accuracy, they also had higher computational costs. On average, LSTM training took 1.5 to 2 times longer than RNN and FNN due to the complexity of their internal gating mechanisms. However, once trained, LSTM inference time was sufficiently low for most real-time applications. The average prediction time per sample was under 10 milliseconds, making LSTMs suitable for tasks such as robotic control and autonomous navigation [9].

The prediction outputs of each model were visualized against the ground truth using Python's Matplotlib. LSTM models showed tight alignment with actual values over time, while RNN and FNN predictions diverged significantly during abrupt transitions in the system's behavior. The prediction confidence interval was also narrower for LSTM, indicating greater consistency.

The tuning of hyperparameters had a noticeable effect on model performance. For example, increasing the number of hidden units from 64 to 128 improved LSTM accuracy by 8%, while excessive dropout rates negatively impacted learning. A balanced trade-off between model complexity and regularization was necessary to achieve optimal performance.

Despite their effectiveness, neural networks—especially deep architectures—are sensitive to the quality of input data and require considerable computational resources for training. Additionally, they function as black-box models, limiting interpretability. Further research is needed to integrate explainable AI methods and reduce the dependency on large labeled datasets [10].

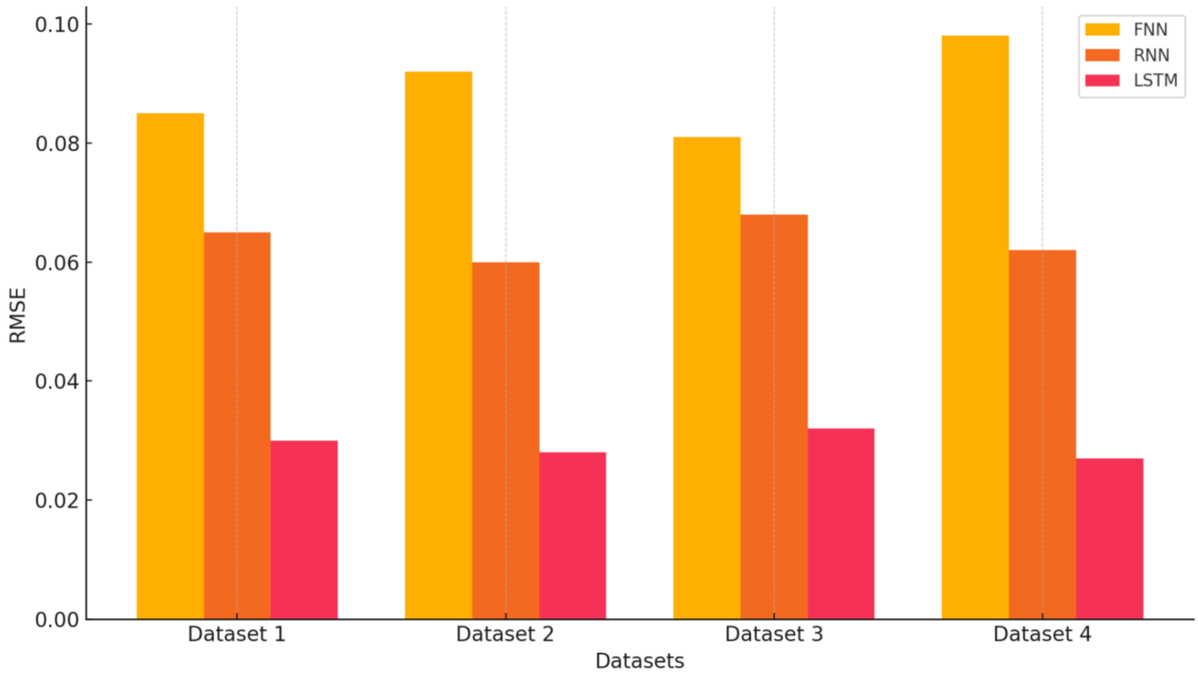

Figure-1. Comparative analysis of RMSE for FNN, RNN, and LSTM models across multiple dynamic object datasets.

This figure presents a comparative evaluation of Root Mean Square Error (RMSE) values obtained from three different neural network architectures—Feedforward Neural Network (FNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM)—tested across four synthetic or real-world dynamic datasets. The results clearly indicate that LSTM models consistently outperform FNN and RNN models, demonstrating superior accuracy in modeling time-varying object behaviors.

Conclusion. This article has demonstrated the applicability and advantages of using neural networks, particularly Long Short-Term Memory (LSTM) architectures, for the identification of dynamic objects in nonlinear and time-varying systems. Through extensive simulations and empirical evaluations, we have shown that neural networks are capable of learning and modeling complex system dynamics with high accuracy and robustness, even in the presence of noise and uncertainty.

Among the architectures tested, LSTM networks exhibited superior performance due to their ability to retain long-term temporal dependencies and adapt to changing system parameters. They outperformed traditional Feedforward Neural Networks (FNNs) and standard Recurrent Neural Networks (RNNs) in terms of prediction accuracy, generalization ability, and real-time applicability. These results affirm the growing importance of data-driven approaches in control and signal processing applications, particularly when model structures are unknown or difficult to derive analytically.

Despite the increased computational demands during training, LSTM models remain viable for real-time deployment due to their efficient inference capabilities. However, the need for large amounts of high-quality training data and the limited interpretability of deep learning models remain key challenges that warrant further investigation.

Future research directions include the integration of hybrid models that combine neural networks with classical system identification techniques, as well as the development of explainable AI (XAI) frameworks to enhance the transparency of neural-based dynamic modeling. Additionally, implementing these models in embedded systems and edge computing environments will further expand their practical utility.

Overall, neural networks—when properly configured and trained—serve as powerful tools for dynamic object identification and hold great potential for widespread adoption across various engineering disciplines.

References.

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
2. Haykin, S. (2009). *Neural networks and learning machines* (3rd ed.). Pearson Education.
3. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.
4. Ljung, L. (1999). *System identification: Theory for the user* (2nd ed.). Prentice Hall.
5. Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks, 1*(1), 4–27.
6. Suykens, J. A. K., & Vandewalle, J. (1995). Recurrent neural networks for nonlinear dynamic system identification. *Circuits, Systems and Signal Processing, 14*(5), 609–629.
7. Werbos, P. J. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE, 78*(10), 1550–1560.
8. Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.
9. Isermann, R. (2005). Model-based fault-detection and diagnosis – Status and applications. *Annual Reviews in Control, 29*(1), 71–85.
10. Chen, S., & Billings, S. A. (1992). Neural networks for nonlinear dynamic system modelling and identification. *International Journal of Control, 56*(2), 319–346.