Special Session: Neuromorphic hardware design and reliability from traditional CMOS to emerging technologies

Liquid Neural Networks vs Incremental Learning in Stock Market Prediction

Hussein Jammal[1,4], Farah Srour[1], Ali Salman[1], Michel Owayjan[1,2], Omran Ayoub[3], Cristina Rottondi[4], Roger Achkar[1]

[1]Faculty of Engineering and Computer Science, American University of Science & Technology, Beriut, Lebanon
[2] Institut de Recherche en Energie Electric de Nantes Atlantique (IREENA), Nantes University, Saint-Nazaire, France
[3]University of Applied Sciences of Southern Switzerland, Switzerland
[4]Politecnico di Torino, Italy
HAJ20010@students.aust.edu.lb, FYS20008@students.aust.edu.lb, ahs20045@students.aust.edu.lb, mowayjan@aust.edu.lb, omran.ayoub@supsi.ch, cristina.rottondi@polito.it, rachkar@aust.edu.lb

***Abstract*—Due to the complexity, challenge, and fast growth of stock markets, they encourage more efficient use of financial resources and the expansion of macroeconomics. Tesla and Apple stock prices fluctuate constantly due to the impact of various industries and market conditions. The extraction of previously unknown patterns and information from time series is central to numerous real-world applications. This paper presents a comparative study of stock market prediction techniques for Tesla and Apple stocks, focusing primarily on the application of Liquid Neural Networks (LNN) and evaluating their performance against Incremental Learning methods. Employing LNN, a novel approach in the field of time-series forecasting, we developed models that leverage the dynamic and flexible structure of LNN to adapt to the complex and non-linear patterns observed in stock prices. The performance of the LNN model was quantitatively assessed using standard metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Directional Accuracy. The LNN model achieved an MSE of 0.000317, RMSE of 0.0178, MAE of 0.014131, MAPE of 1.8%, and a Directional Accuracy of 49.36%. In contrast, the Incremental Learning approach yielded an MSE of 0.54355, RMSE of 0.73725, MAE of 0.73158, MAPE of 89.39%, and Directional Accuracy of 28.37%. These results underscore the superior accuracy and effectiveness of LNNs in modeling stock market behaviors compared to traditional Incremental Learning methods. The findings not only demonstrate the potential of LNNs in financial analytics but also suggest avenues for further research into enhancing predictive accuracy and model robustness.**

Keywords— Liquid Neural Networks (LNN), Incremental Learning, Stock Market Prediction, Time-Series Forecasting, Financial Analytics, Deep Learning, Market Behavior Analysis, Predictive Modeling, Tesla Stock, Apple Stock.

I. INTRODUCTION

The stock market plays a role in economic progress. The stock market has gained popularity among institutions and investors due to its high return qualities. However, the stock market's intricate volatility can often result in massive losses for institutions or investors. The stock market's prediction and forecast of stock price changes might help investors minimize risk.

There has been significant progress in research on the role of machine learning in predicting stock prices. In the field of machine learning, algorithms create a mathematical model using sample data, which is sometimes referred to as "training data," to make predictions or judgments, such as Tesla's and Apple's Markets [1]. Algorithms examine large amounts of data, discovering subtle market patterns and predicting future stock values. Researchers employ a variety of criteria, including historical data and temporal trends, to test current ideas and find patterns in the stock market. Studies use machine learning techniques like supervised learning to predict stock prices, as well as methodologies like deep learning and non-linear regression.

Incremental learning is a machine learning methodology where a model continuously learns new information over time, enhancing its performance without retraining from scratch. In stock market prediction, incremental learning techniques adapt to evolving market conditions, making them particularly useful for predicting volatile assets. Specifically, in stock market prediction, incremental learning algorithms can adjust to real-time data streams, capturing sudden market shifts and complex patterns [2]. This adaptability enables more accurate and timely predictions, which is crucial for investors navigating dynamic market environments.

Liquid Neural Networks (LNNs) represent a novel approach to prediction tasks, using their time-continuous recurrent architecture to process sequential data dynamically. In prediction processes, LNNs exhibit remarkable capabilities owing to their unique characteristics. They offer adaptability and fast adjustment to changing data patterns, thus enhancing prediction accuracy and robustness. This dynamic adaptability enables LNNs to effectively capture temporal dependencies in the data, resulting in more precise predictions across various domains such as weather prediction, speech recognition, and autonomous driving. LNNs offer several advantages in prediction processes. The characteristic architecture of LNNs allows for real-time adjustments to changing data patterns, ensuring robust performance even in dynamic environments. Thus, LNNs emerge as a powerful tool for predictive analytics, offering enhanced adaptability, efficiency, and accuracy.

Fig. 1. Stock Market Close Price History

In the context of predicting the stock market, an LNN would analyze historical stock market data, including factors such as past stock prices, trading volumes, market sentiment, and relevant economic indicators. The network would then process this data through its dynamic architecture, which allows for the capture of temporal dependencies and patterns within the stock market data shown in Fig. 1. By analyzing these temporal relationships, LNN can identify trends and correlations that may affect the stock price in the future. Additionally, LNNs can adapt and adjust their internal states based on incoming data, enabling them to continuously update their predictions as new information becomes available. This adaptability is crucial in the stock market prediction field, where market conditions can change rapidly. Overall, through its dynamic processing capabilities and adaptability, a Liquid Neural Network (LNN) can effectively capture the complexities of the Stock Market and generate accurate predictions.

This paper presents a comparative analysis of these Liquid Neural Network and Incremental Learning methodologies applied to the prediction of Tesla and Apple stock prices. We chose Tesla and Apple due to their high volatility and investor interest, which make them ideal candidates for testing advanced predictive models. By implementing and comparing LNNs with Incremental Learning approaches, this study aims to uncover insights into their applicability, strengths, and limitations within the scope of financial market predictions. The evaluation of these models is conducted using a comprehensive set of metrics including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Directional Accuracy. These metrics were selected to provide a thorough understanding of each model's predictive power and accuracy, addressing both the magnitude and direction of predictions, which are critical for investment decisions.

The rest of the paper is organized as follows. Section II discusses related work in the field of stock market prediction and the specific technologies of LNN and Incremental Learning. Section III describes the methodology, including data collection, model architecture, and evaluation. Section IV discusses the experimental results. Section V concludes with a summary of findings and potential areas for future research.

II. LITERATURE REVIEW

A. *Incremental Learning*

Incremental Learning utilizes two types of methods: incremental learning, in which the model updates with every single gathered current stock price from the data stream, and Offline-Online learning, in which the model is retrained after each trading session. Incremental Linear Regression has been used for incremental models, whereas Offline-Online models have been predicted using LSTM and CNN variations [3]. In the Offline-Online technique, offline means studying a batch of data and improving the model to produce a prediction, whereas online means extracting samples from streaming data and generating the prediction. In contrast, incremental learning aims to create a learning model that adapts to new input while retaining all prior information. The Offline-Online technique does not fine-tune the model upon receiving each new instance from the stream; rather, it is modified after each trading session because the stock market might be influenced by various things throughout a session. The incremental learning model changes with each stream instance, eliminating the need for retraining with the full dataset [4].

The implementation of the incremental learning model may be divided into many steps, including data collection, processing, feature selection, and the incremental learning model itself. A simple flowchart is provided below the incremental learning model procedure in Fig. 2. Recent advancements have integrated deep learning with Incremental Learning to manage real-time data adjustments effectively. Singh et al. (2023) [5] demonstrated the feasibility of using deep networks for dynamic task adaptation, which is pertinent to adapting financial models to market changes. In the financial domain, Incremental Learning has been used to refine predictions continuously as new market data becomes available, ensuring models remain robust over time.

B. *Liquid Neural Network (LNN)*

In 2020, a group of MIT scientists produced a study titled "Liquid Time-constant Network" in which they introduced the concept of liquid neural networks. This concept is influenced by brain research. They investigated the underlying computation of a tiny worm, C.elegans, whose nervous system has just 302 neurons yet generates adaptive dynamics. Liquid neural networks are exciting because they can adapt to changing situations, rather than simply during training. They are especially effective for processing time series data, making them ideal for applications like autonomous driving and medical diagnostics.

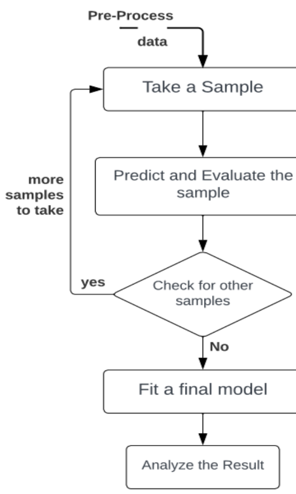

Fig. 2. Incremental Learning Process Flowchart

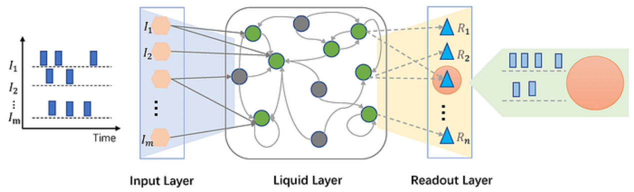

Fig. 3. Liquid Neural Network Architecture

Liquid Neural Networks (LNN) have emerged as a promising new frontier. It is a time-continuous system that processes data sequentially, remembers previous inputs, modifies its behaviors depending on new inputs, and can handle variable-length inputs to improve NNs' task-understanding skills. LNN design varies from typical neural networks in that it can efficiently process continuous or time series data. If fresh data becomes available, LNNs can adjust the number of neurons and connections per layer. Liquid Neural Networks (Fig. 3) are designed specifically for time series data processing and prediction [6].

$$\frac{dx(t)}{dt} = -\left[\frac{1}{\tau} + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta)\right]\mathbf{x}(t) + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta)A \qquad (1)$$

The above equation is derived by (Funahashi and Nakamura 1993) and may be used to determine a more stable continuous-time recurrent neural network:

$$\text{(CT-RNN): } \frac{dx(t)}{dt} = -\frac{\mathbf{x}(t)}{\tau} + f\,(\boldsymbol{x}(t), \boldsymbol{I}(t), t, \theta) \qquad (2)$$

where the phrase $-\frac{x(t)}{\tau}$ helps the autonomous system attain an equilibrium state with a time constant τ. $\boldsymbol{x(t)}$ is the hidden state, $\boldsymbol{I(t)}$ is the input, t represents time, and f is parametrized by θ.

Offering an alternative formulation: let the hidden state flow of a network be proclaimed by a system of linear ODEs of the form:

$$\frac{dx(t)}{dt} = -\frac{\mathbf{x}(t)}{\tau} + S(t) \qquad (3)$$

Let S(t) $\in \mathbb{R}^M$ represent the following nonlinearity specified, with parameters θ and A [6].

$$S(t) = f\,(\boldsymbol{x}(t), \boldsymbol{I}(t), t, \theta)(A - \boldsymbol{x}(t)), \qquad (4)$$

C. Related Work

For the past 35 years, we have developed probabilistic models that make predictions based on data and learned parameters. Each neuron acts as a logistic regression unit. By using backpropagation, which adjusts the neuron's parameters to minimize errors, the network learns and improves its predictions. Neural networks (Fig. 4) face certain challenges today; they excel in unified tasks, but cannot generalize knowledge across tasks. Additionally, they process data non-sequentially, making them inefficient at handling real-time data.

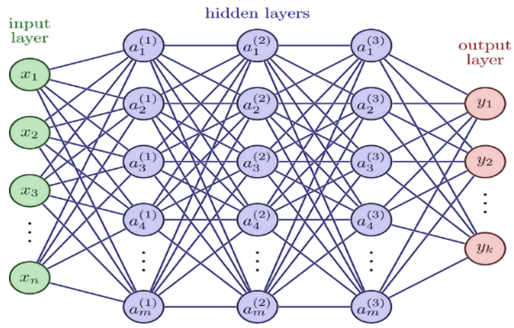

Fig. 4. General Neural Network Architecture

The field of stock market prediction has long been a rich area of academic and practical interest, combining economic theory with advanced computational techniques. The seminal work by Thakkar and Chaudhari (2021) [7] on time series analysis has influenced numerous studies on market prediction techniques. More contemporary approaches have integrated machine learning methods to improve prediction accuracy and handle larger datasets.

Notably, the application of neural networks to stock market prediction has been explored in various studies. Zhang et al. (2017) [8] demonstrated the use of deep learning for predicting stock prices with significant success over traditional statistical methods. Their work has been pivotal in showing the efficacy of neural networks in capturing the complex patterns of financial markets.

Recent research has also focused on the application of novel neural network architectures and hybrid models. For instance, Dixon et al. (2016) [9] combined machine learning algorithms with quantitative finance models to predict stock returns, offering insights into the integration of traditional financial theories with modern computational techniques.

III. Architecture & Methodology

This section outlines the methodology employed in this study, encompassing data collection, exploratory data analysis (EDA), feature engineering, model architecture, and evaluation metrics. Our objective is to leverage historical stock data from Yahoo Finance for Apple (AAPL) and Tesla (TSLA) to predict future stock prices using advanced machine learning techniques.

A. Data Collection

The dataset consists of daily stock prices for AAPL and TSLA retrieved from Yahoo Finance using the Python library **yfinance**. The data spans from January 1, 2010, for AAPL, and from June 29, 2010, for TSLA, up to the day prior to the current date. Each dataset includes the following features: Open, High, Low, Close, Adjusted Close, and Volume.

B. Exploratory Data Analysis (EDA)

Initial data exploration was conducted to understand underlying patterns and distributions:

1. Candlestick Plot with Moving Averages (20, 50, 100 days): Visualizes price movements over time alongside key moving averages.

2. Correlation Heatmap: Assesses the relationships between different stock attributes.
3. Box Plot: Identifies outliers within the dataset.
4. Histogram of Daily Price Changes: Examines the distribution of daily price fluctuations.

C. *Feature Engineering*

To enhance model performance, several technical indicators were computed, resulting in a total of 28 attributes [10]. Table I presents the feature engineering done on the model.

The equations for each attribute are found in [11].

Normalization was applied using MinMaxScaler to scale all input features to a [0, 1] range, facilitating model convergence and improving training stability. The MinMaxScaler method is chosen for the normalization of the input dataset, and the formula is as follows [12]:

$$X_{new} = \frac{X_{orginal} - X_{\min}}{X_{max} - X_{\min}} \quad (5)$$

D. *Dataset Construction*

The training and testing datasets are constructed using a **look_back** parameter of 10 days, creating sequences from the time series data that represent the past 10 days of trading to predict the next day's adjusted close price [13].

The dataset is split into 80% for training and 20% for testing to evaluate the models' performance.

E. *Model Architecture*

1) *Liquid Neural Networks (LNN)*

The LNN model utilized the LTCCell class, configured with 150 units incorporating dropout and regularization (L1 and L2) to prevent overfitting and enhance generalization [14].

2) *Incremental Learning (MLP)*

The MLP architecture consisted of three dense layers with 256, 128, and 64 units respectively, followed by a dropout layer and a final dense layer outputting the predicted adjusted close price. The activation function used for the hidden layers is the **Relu** activation function.

F. *Evaluation Metrics*

The models were assessed using the following metrics:

1. Mean Squared Error (MSE): Measures the average squared difference between the actual values and the predicted values. It penalizes larger errors more than smaller ones due to the squaring. A lower MSE indicates a better fit of the model to the data.

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (6)$$

2. Root Mean Squared Error: This is the square root of the MSE. It is in the same units as the original data, making it more interpretable. It also penalizes larger errors more heavily. Lower RMSE values indicate better model performance.

TABLE I. FEATURE ENGINEERING DONE ON THE MODEL

Attributes		***Target Output***
Open, High, Low, Close, Volume	14 & 21 Day Moving Average Convergence Divergence	Adjusted Close
Daily Returns	Pivot Point	
5-Day Momentum	On Balance Volume	
14-Day Bollinger Bands	14-Day Average True Range	
14-Day Fast, Slow & Smoothed Slow Stochastic Indicators	7, 14 & 21 Day Up & Down Trending Fibonacci Retracement at 38.2, 50 & 61.8%	
Past 8 Weekly & Past 2 Monthly Returns	3 Day Rate of Change	
14 Day Simple & Exponential (k=2/5) Moving Average	14-Day Relative Strength Index	

$$\sqrt{\sum_{i=1}^{n}\frac{(y_i - \hat{y}_i)^2}{n}} \quad (7)$$

3. Mean Absolute Error (MAE): Measures the average absolute difference between the actual values and the predicted values. It provides a linear score which means all individual differences are weighted equally. Lower MAE values indicate better predictive accuracy.

$$\frac{1}{n}\sum_{i=1}^{n}| y_i - \hat{y}_i| \quad (8)$$

4. Mean Absolute Percentage Error (MAPE): Measures the average absolute percentage difference between the actual values and the predicted values. It expresses the accuracy as a percentage, which is useful for comparing model performance across different scales. Lower MAPE values indicate better model performance. However, MAPE can be problematic when actual values are very small or zero.

$$\frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \quad (9)$$

5. Directional Accuracy: Measures the proportion of times the predicted change direction matches the actual change direction. It evaluates the model's ability to correctly predict the direction of change. A higher directional accuracy indicates a better model in terms of predicting the trend correctly [15].

$$\frac{1}{n}\sum_t 1_{sign(y_t - y_{t-1}) == sign(\hat{y}_t - y_{t-1})} \quad (10)$$

Where, y_i and $\hat{y}_i$ in each metric indicate the actual price and the predicted price at time t, respectively; n represents the number of observations in the dataset.

IV. Result & Explanation

A. User Interface

A user interface was developed for the stock market prediction application. The application leverages Liquid Neural Network (LNN) and Incremental Learning techniques to predict stock prices [16].

B. Results

Fig. 5 represents predictions of Apple & Tesla stock closing prices using the Liquid Time-Constant (LTC) model. It effectively demonstrates the efficacy of the LTC model in predicting different stock prices. The close fit between the predicted values and the actual historical prices in both the training and testing phases suggests that the model is well-calibrated and capable of making accurate stock market predictions. This result is a key outcome of the analysis presented in the Liquid Neural Network (LNN) presentation.

The model also captures the overall trend in the testing phase but exhibits some inaccuracies during volatile periods. These results indicate that the LTC model is effective in predicting general trends in stock prices. This analysis aligns with the objectives and findings presented in the Liquid Neural Network (LNN) presentation.

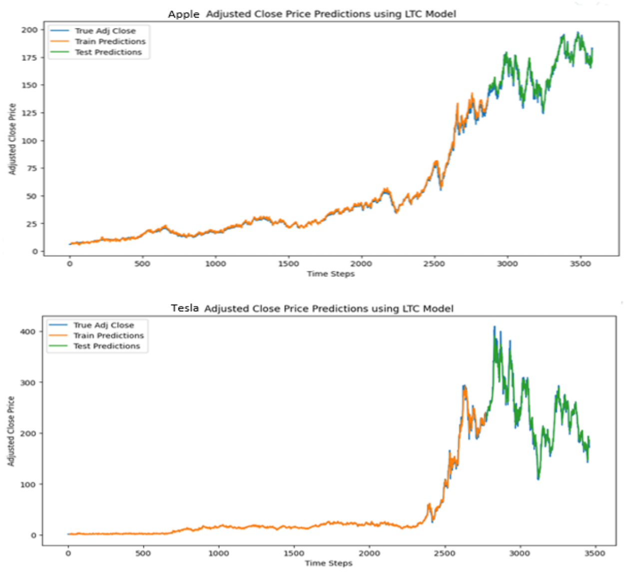

Fig. 5. Apple & Tesla Stock Market Predictions

C. Common Results

- X-axis (Time Steps): Represents the sequential time steps, which could correspond to days, weeks, or any other time interval used in the dataset.
- Y-axis (Adjusted Close Price): Represents the adjusted closing price of stock.
- Blue Line (True Adj Close): This line shows the actual historical adjusted closing prices of each stock.
- Orange Line (Train Predictions): This line shows the model's predictions for the training dataset, indicating how well the model fits the historical data it was trained on.
- Green Line (Test Predictions): This line shows the model's predictions for the testing dataset, which is used to evaluate the model's performance on unseen data.

D. Loss Function

The graphs in Fig. 6 depict the training and validation loss for models predicting the adjusted close prices of Apple and Tesla stocks. The loss functions are plotted over the epochs, which are iterations of training the model on the entire training dataset.

Both the Apple and Tesla models demonstrate effective learning, as shown in Fig. 6, by the significant reduction in loss from the beginning to the end of training. The close alignment of training and validation losses suggests good generalization, indicating the models are not overfitting to the training data. The loss function graphs confirm that both models have been effectively trained, with low and stable losses across both datasets. Although Tesla's model encounters slightly more challenges due to higher volatility, the overall performance suggests these models are robust for making stock price predictions.

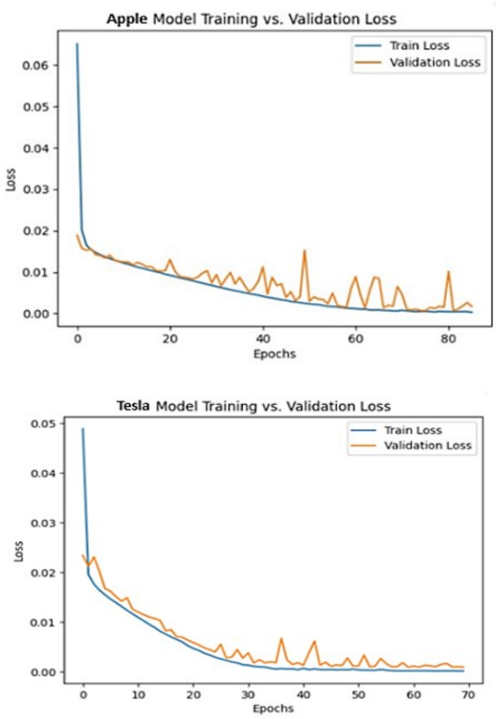

Fig. 6. Apple & Tesla Model Training Loss

E. Evaluation Metrics

Table II evaluates the performance of two predictive models—LTC Model, and Incremental Learning—on Tesla's stock market metrics. Across all evaluated metrics, the LTC Model significantly outperforms the others, demonstrating the lowest Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE), which indicates it predicts having a minimal error. It also has the lowest Mean

Absolute Percentage Error (MAPE) at 4.87%, showing highly accurate percentage predictions. Additionally, it achieves the highest Directional Accuracy (DA) at 51.25%, making it the most effective model in predicting the direction of stock price movements.

Table III summarizes the performance metrics of two models used for predicting apple prices. The LTC Model excels in most metrics, demonstrating the lowest Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), indicating highly accurate predictions with minimal errors in both magnitude and percentage terms.

TABLE II. EVALUATION METRICS FOR TESLA STOCK MARKET

Metrics	Models	
	LTC Model	***Incremental Learning***
Mean Squared Error (MSE)	0.002488	0.337814
Root Mean Squared Error (RMSE)	0.04988	0.581218
Mean Absolute Error (MAE)	0.03179	0.561843
Mean Absolute Percentage Error (MAPE)	4.87%	94.79%
Directional Accuracy (DA)	51.25%	14.68%

TABLE III. EVALUATION METRICS FOR APPLE STOCK MARKET

Metrics	Models	
	LTC Model	***Incremental Learning***
Mean Squared Error (MSE)	0.00317	0.54355
Root Mean Squared Error (RMSE)	0.0178	0.73725
Mean Absolute Error (MAE)	0.014131	0.73158
Mean Absolute Percentage Error (MAPE)	1.80%	89.39
Directional Accuracy (DA)	49.36%	28.37%

Overall, the LTC Model consistently shows superior performance across all metrics, making it the most effective model for predicting Tesla and Apple stock market metrics compared to the Incremental Learning models.

V. CONCLUSIONS AND FUTURE WORK

A. Conclusions

This study demonstrates the application and comparative effectiveness of Liquid Neural Networks (LNN) and Incremental Learning methods in predicting the stock prices of Tesla and Apple. LNN outperformed incremental learning in almost all tested metrics, offering a more accurate and reliable forecasting tool for stock market investors. LNN's adaptability and dynamic structure make it exceptionally suited for capturing the complex, nonlinear patterns observed in financial markets like those of Tesla and Apple. The study's findings emphasize the potential of Liquid Neural Networks not only in enhancing the prediction but also in increasing the robustness of financial analytics tools. As markets continue to evolve, the adaptability and precision of models like LNNs will be crucial in developing tools that can leverage vast amounts of data for real-time decision-making, ultimately benefiting the broader field of financial technology.

B. Future Work

The promising results obtained from employing Liquid Neural Networks (LNN) for stock market prediction establish a strong foundation for several future research directions. First, further exploration into hybrid models that integrate LNN with other advanced machine learning techniques could potentially enhance prediction. Second, expanding the scope of the data used for training the models could significantly impact their effectiveness. Incorporating more diverse datasets, including macroeconomic indicators, social media sentiment analysis[17], or geopolitical events [18], could provide more comprehensive inputs that reflect the multitude of factors influencing stock prices. Third, experimenting with different configurations of the LNN architecture, such as varying the density and connectivity of the liquid states or exploring different types of regularization, could uncover more optimal settings for specific types of stock market data or prediction horizons.

Additionally, real-time data streaming and incremental model updates represent a crucial area for development. Implementing a system that continuously updates its predictions based on new data could be highly beneficial for high-frequency trading strategies. Finally, exploring the interpretability of LNN models could help in understanding how these models make their predictions, which is vital for trust and adoption in real-world applications. Techniques such as visualization of the liquid states could provide insights into the model's decision-making process. These future research efforts could improve the performance and applicability of LNN models in stock market prediction and contribute to the field of financial analytics by providing tools that can handle the complexity and dynamic nature of financial markets.

REFERENCES

[1] Polanki, K., Roa Purimetla, M., Roja D, Khader Basha Sk., Thommandru, R., & Javvadi, S. (2023, December 18). Predictions of Tesla Stock Price Based on Machine Learning Model.

[2] Mahtab, M. T., Zaman, A. G. M., Mahin, M. R., Mia, M. N., & Islam, M. T. (2022, December 31). Stock Price Prediction: An Incremental Learning Approach Model of Multiple Regression. the AIUB Journal of Science and Engineering.

[3] Botunac, I., Bosna, J., & Matetić, M. (2024, February 28). Optimization of Traditional Stock Market Strategies Using the LSTM Hybrid Approach.

[4] Zhao, L., Kong, S. and Shen, Y., 2023, August. DoubleAdapt: A Meta-learning Approach to Incremental Learning for Stock Trend Forecasting. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 3492-3503).

[5] Singh, T., Kalra, R., Mishra, S., Satakshi and Kumar, M., 2023. An efficient real-time stock prediction exploiting incremental learning and deep learning. Evolving Systems, 14(6), pp.919-937.

[6] Hasani, R., Lechner, M., Amini, A., Rus, D. and Grosu, R., 2021, May. Liquid time-constant networks. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 9, pp. 7657-7666).

[7] Thakkar, A. and Chaudhari, K., 2021. A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions. Expert Systems with Applications, 177, p.114800.

[8] Zhang, L., Aggarwal, C., & Qi, G.-J. (2017). Stock Price Prediction via Discovering Multi-Frequency Trading Patterns. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[9] Dixon, M., Klabjan, D., & Bang, J. H. (2016). Classification-based Financial Markets Prediction using Deep Neural Networks. Algorithmic Finance, 6(3-4), 67-77.

[10] Sethia, A. and Raut, P., 2019. Application of LSTM, GRU and ICA for stock price prediction. In Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2018, Volume 2 (pp. 479-487). Springer Singapore.

[11] Thakkar, A. and Chaudhari, K., 2020. Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks. Applied Soft Computing, 96, p.106684.

[12] Zhang, H., Lin, H. and Li, Y., 2015. Impacts of feature normalization on optical and SAR data fusion for land use/land cover classification. IEEE Geoscience and Remote Sensing Letters, 12(5), pp.1061-1065.

[13] Liu S., Liao G., & Ding Y., (2018). Stock transaction prediction modeling and analysis based on LSTM. In 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). IEEE; p. 2787–2790.

[14] R. H. Memisevic, "liquid_time_constant_networks," GitHub repository, 2023. [Online]. Available: https://github.com/raminmh/liquid_time_constant_networks

[15] Wikipedia contributors, Mean directional accuracy – Wikipedia, The Free Encyclopedia, 2020, https://en.wikipedia.org/w/index.php?title=Mean_directional_accuracy&oldid=956009191; [Online; accessed 17-July-2020].

[16] H. Jammal, "Liquid-Neural-Networks-in-Stock-Market-Prediction," GitHub repository, 2024. [Online]. Available: https://github.com/HusseinJammal/Liquid-Neural-Networks-in-Stock-Market-Prediction

[17] Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine learning-based sentiment analysis for twitter accounts. Mathematical and computational applications, 23(1), 11.

[18] Chaudhary, A., Gupta, A., Pahariya, D., & Singh, S. (2023, January 1). Stock Price Prediction of Tesla & Apple using LSTM. ITM Web of Conferences.