Real Estate Price Prediction Using Deep Learning Models: A Comparative Study on the Zillow Dataset

Shailesh K R
School of CSE & IS,
Presidency University,
Bengaluru-560064,India
krshailesh627@gmail.com

Pakruddin .B
School of CSE & IS,
Presidency University,
Bengaluru-560064, India
fakrubasha@gmail.com

Syed Afridi
School of CSE & IS,
Presidency University,
Bengaluru-560064, India
syedafridi644@gmail.com

Praveen Nandan K
School of CSE & IS,
Presidency University,
Bengaluru-560064, India
praveennandu4@gmail.com

Abhinav K
School of CSE & IS,
Presidency University,
Bengaluru-560064, India
abhinavkesavan@gmail.com

Shubam V Patil
School of CSE & IS,
Presidency University,
Bengaluru-560064, India
sopatil.2003@gmail.com

Abstract*—This study presents a comparative analysis of advanced deep learning models for real estate price prediction using the Zillow dataset, addressing the limitations of traditional methods in capturing complex, non-linear dependencies in time series data. We evaluate state-of-the-art architectures, including Neural Temporal Models (NeuTS), TimeGPT, Temporal Convolutional Networks (TCNs), CNN-LSTM hybrids, and Deep Autoregressive models (DeepAR), to effectively model both short-term fluctuations and long-term trends. A robust framework is developed, incorporating data preprocessing, feature engineering, hyperparameter tuning, and performance evaluation using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 score. Additionally, an interactive web application built with Streamlit is introduced to visualize and interpret forecasting results in real time, enhancing stakeholder engagement and decision-making. The study highlights the superior accuracy and robustness of deep learning models, offering practical insights for real estate market analysis and investment strategies.

Keywords*— *Real-Estate Price Prediction; Deep Learning; NeuTS; TimeGPT; TCN; CNN-LSTM;

I. INTRODUCTION

Robust time series forecasting is fundamental across various domains, including real estate price prediction, financial markets, and climate modeling. Traditional statistical models like ARIMA and exponential smoothing often struggle to capture the intricate non-linear dependencies inherent in real-world data. Recent advancements in deep learning have revolutionized forecasting [11] by leveraging neural networks to model complex temporal relationships.

Traditional methods like ARIMA and exponential smoothing often fail to capture complex, non-linear patterns, while deep learning models such as NeuTS, TimeGPT, Temporal Convolutional Networks (TCNs), CNN-LSTM, and DeepAR have emerged as powerful tools for addressing these challenges. Despite advancements, challenges like the need for large datasets, irregularities, noise, and missing values persist. Deep learning models like NeuTS [9] and TimeGPT [3] automatically learn features from raw data, reducing the need for extensive feature engineering, but they require careful tuning and can be computationally intensive .

Our proposed framework integrates models like NeuTS, TimeGPT, TCNs, CNN-LSTM, and DeepAR to capture both short-term and long-term dependencies. NeuTS and TimeGPT use transformer architectures, TCNs employ convolutional layers, CNN-LSTM [2] helps in feature extraction, temporal learning, sequence prediction, optimization. and DeepAR provides probabilistic forecasts [4].

Our research demonstrates the superior performance of these deep learning models in time series forecasting, showing high accuracy and robustness in capturing complex temporal patterns. The contributions include a comprehensive framework, identification of key features, and optimization of predictive models, aiding stakeholders in making informed decisions [18].

II. LITERATURE REVIEW

Deep learning has revolutionized the field of time series forecasting by offering advanced techniques that surpass traditional statistical methods. Recently developed Models such as NeuTS [9] and TimeGPT [3] have demonstrated significant potential in capturing complex temporal dependencies through the use of transformer architectures, enabling more accurate predictions. Temporal Convolutional Networks (TCNs) [7] have proven effective in managing sequential data with convolutional layers, delivering robust forecasts across diverse applications. Furthermore, CNN-LSTM [1] and DeepAR [5] enhance forecasting capabilities by addressing high-dimensional data and providing probabilistic predictions, respectively, thereby improving reliability and interpretability. These advancements bridge the gap between traditional methods and the need for more precise and adaptable forecasting tools [18], underscoring the potential of deep learning in enhancing the accuracy and reliability of time series forecasting [11]. The ability of these models to capture long-term dependencies and complex patterns has made them indispensable for forecasting tasks in fields such as finance, healthcare, supply chain management and particularly in house price prediction. Their adaptability to irregular data patterns and scalability to large datasets have further strengthened their applicability in real-world scenarios, positioning deep learning as a cornerstone for modern time series forecasting.

III. PROPOSED METHODOLOGY

A. Dataset used:

We used the Zillow dataset [20], which includes: The dataset contains 895 rows and 305 columns. Table I describe sample from dataset.

TABLE I. SAMPLES FROM DATASET

Column Name	**Description**
RegionID	Unique identifier for each region.
SizeRank	Rank of the region based on size/population
RegionName	Name of the region (city, county, metro area, etc.).
RegionType	Type of region (e.g., metro, county, ZIP code).
StateName	U.S. state where the region is located.
Date Columns	Home values recorded monthly from January 2000 to December 2024

B. Data Preprocessing:

The dataset was preprocessed using the melting technique in Pandas, converting it from wide to long format for efficient time series analysis and handling missing values. Example Syntax:- pd.melt(df, id_vars, value_vars, var_name, value_name).

C. Model Architectures:

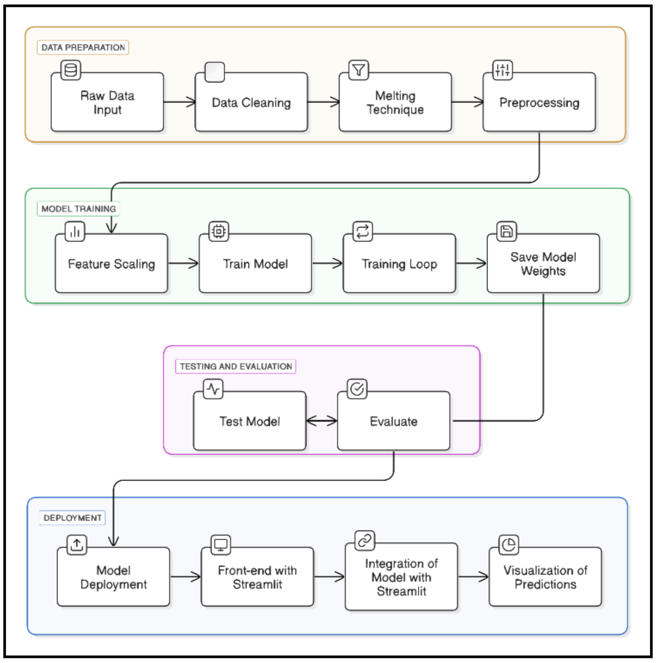

Fig.1: Overview of the Deep Learning Pipeline Architecture

The deep learning pipeline Fig.1 is systematically organized into four pivotal stages: Data Preparation, Model Training, Testing and Evaluation, and Deployment. The data preparation phase involves meticulous cleaning, transformation, and preprocessing to handle missing values, outliers, and inconsistencies, ensuring data quality and consistency. Feature scaling and encoding techniques are employed to normalize data distributions and improve model convergence. During model training, optimization algorithms such as gradient descent and backpropagation are utilized to minimize loss and enhance predictive accuracy. Hyperparameter tuning is conducted to refine model performance. The testing and evaluation phase involves rigorous assessment using cross-validation and performance metrics such as RMSE and MAE to ensure generalizability and mitigate overfitting. The deployment phase integrates the trained model into a Streamlit-based interface, facilitating real-time inference and user interaction. This structured framework ensures a seamless transition from data acquisition to model deployment, promoting scalability and operational efficiency.

The Various models that can be employed in this pipeline include:

1. NeuTS Model:

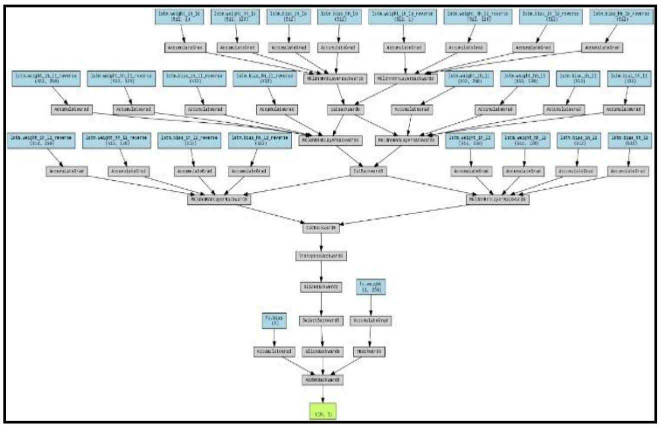

Fig.2: NeuTS Model Architecture

The NeuTS model [9, 10] is an LSTM-based neural network for time-series forecasting. It consists of: An LSTM layer with a hidden dimension of 64 and two layers for sequential pattern learning. A fully connected (FC) layer that maps the final LSTM output to the desired prediction output as shown in Fig.2. In the forward pass, the input sequence is processed through the LSTM, and the last time step's output is passed through the FC layer to generate the final prediction.

2. TimeGPT Model:

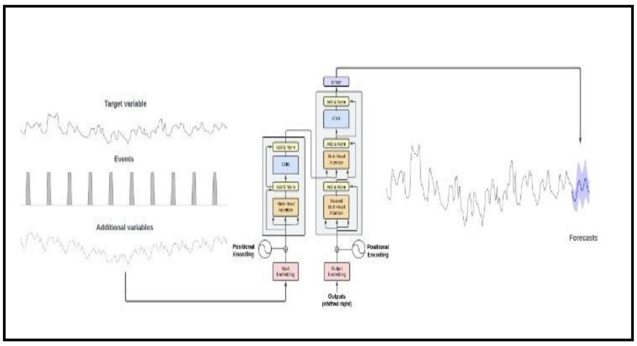

Fig.3: TimeGPT Model Architecture and Forecasting Process

The Fig.3 illustrates the architectural framework and forecasting mechanism of the TimeGPT model. It depicts the input processing of a target variable, event-based signals, and additional covariates, which are fed into a series of transformer blocks for sequential feature extraction and temporal dependency modeling.

The TimeGPT model [3] is a transformer-based AI model designed for time-series forecasting. It features a pre-trained transformer backbone optimized for long-term forecasting tasks, ensuring accurate predictions over extended periods. The model employs sliding window cross-validation to enhance robustness and generalization across different datasets. Additionally, it provides confidence interval estimations (80%, 90%, and 99.5%) to quantify forecast uncertainty.

A key advantage of TimeGPT model [4] is its zero-shot forecasting capability, which enables it to generalize to unseen time-series data without requiring task-specific fine-tuning, leveraging extensive pre-training on diverse datasets. During the forward pass, the input time-series data is processed through the transformer-based architecture, generating forecasts with prediction intervals to estimate uncertainty.

3. TCN Model:

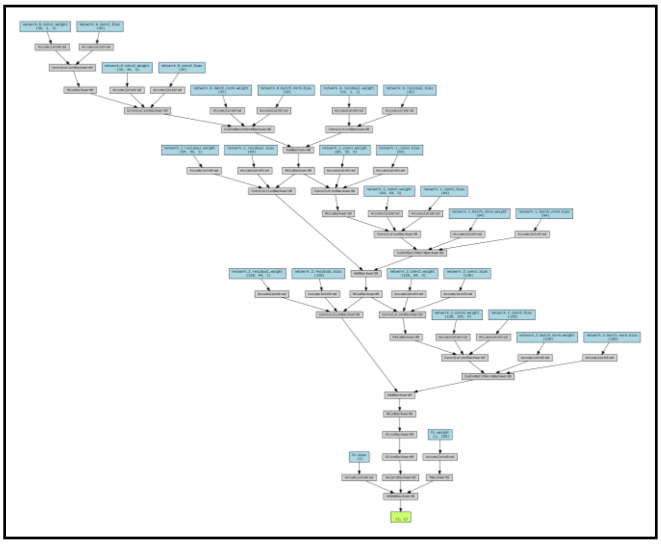

Fig.4: TCN Model Architecture

The Temporal Convolutional Network (TCN) model [7, 8] is a convolutional-based deep learning model for time-series forecasting. It consists of: Multiple TCN blocks, each containing dilated causal convolutions to capture long-range dependencies. ReLU activation after each convolutional layer to introduce non-linearity. A fully connected output layer that maps the final convolution output to a prediction. In the forward pass, the input is processed through stacked TCN blocks, followed by a final FC layer to generate the forecasted value. The Fig.4 shows the parallel nature of convolutional operations, where intermediate feature maps are processed independently and aggregated through skip connections and combination layers.

4. CNN-LSTM Model:

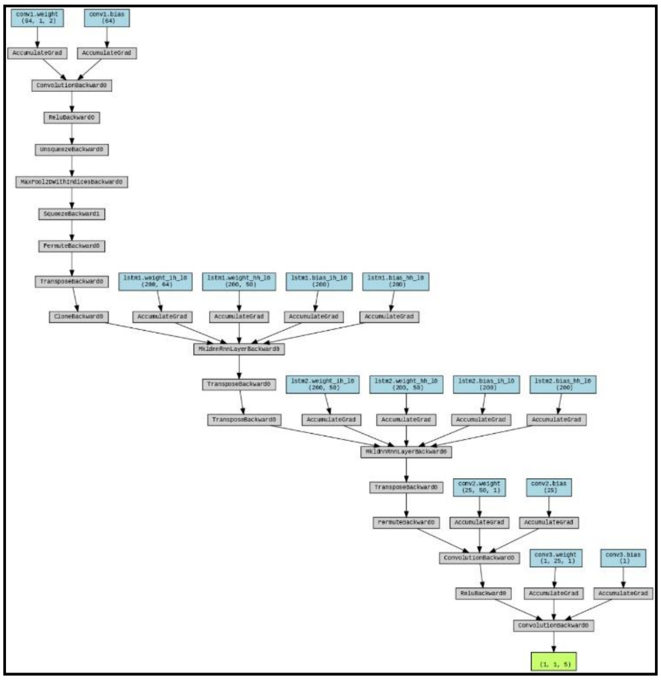

Fig.5: CNN-LSTM_Model Architecture

The CNN-LSTM model [1, 2] is a hybrid neural network for time-series forecasting. It consists of: A Conv1D layer with filters=64 and kernel size=3 to extract local patterns. A MaxPooling1D layer to reduce dimensionality. The upper section as shown in Fig.5 represents the convolutional layers (CNN), where input data is processed through a series of convolution and pooling operations to extract spatial features. These feature maps are then passed to the LSTM layers for sequential modeling.
An LSTM layer with num_layers=2 and hidden_size=64 to model sequential dependencies. A fully connected (FC) output layer that maps the last LSTM hidden state to a single-step prediction. In the forward pass, the input sequence is processed through the Conv1D and MaxPooling1D layers, followed by the LSTM layers. The final hidden state is passed through the FC layer to predict the next time step.

5. DeepAR Model:

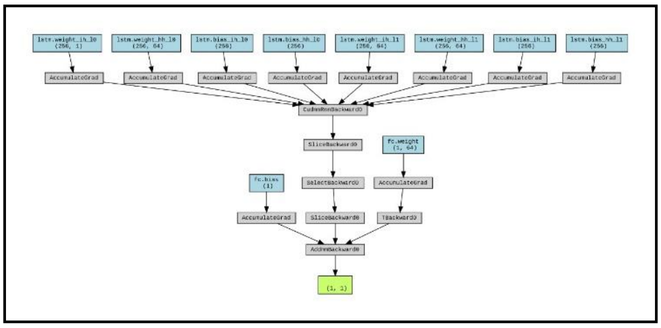

Fig.6: DeepAR Model Architecture.

The DeepAR [5, 6] model is an autoregressive LSTM-based neural network for probabilistic time-series forecasting. It consists of: An LSTM layer with num_layers=2 and hidden_size=64 to model sequential dependencies. A fully connected (FC) output layer that maps the last LSTM hidden state to a single-step prediction. In the forward pass, the input sequence is processed through the LSTM layers, and the final hidden state is passed through the FC layer to predict the next time step. The CombinedBackward operation in Fig.6, which aggregates gradients during backpropagation to facilitate efficient weight updates.

D. Training and Evaluation:

- Training: Split the dataset into 80%-training and 20%-test sets.
- Evaluation Metrics:

a) MAE=(1/n) * Σ |(y_i - ŷ_i)|

b) MSE=(1/n) * Σ (y_i - ŷ_i)^2

c) R^2 Score=(1/n) * Σ (y_i - ŷ_i)^2

(where "y_i" is the actual value, "ŷ_i" is the predicted value, and "n" is the number of data points)

TABLE III. HYPERPARAMETER TUNING

Model Name	Learning Rate (η)	Batch Size	Number of Epochs
NeuTs	0.0005	Null (implicitly uses the entire dataset)	100
TimeGPT	Not applicable as it leverages pre-trained capabilities.		
TCN	0.001	64	20
CNN-LSTM	0.001	16	50
DeepAR	0.001	64	20

IV. EXPERIMENTAL RESULTS

The experimental results highlight the performance of various deep learning models—NeuTS, TimeGPT, TCN, CNN-LSTM , and DeepAR—in predicting real estate prices using the Zillow dataset. The models were evaluated based on Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 score, with the results summarized in Table I and visually represented in Fig. 7.

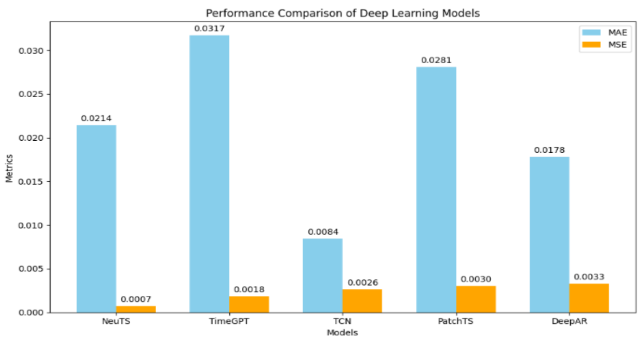

Fig.7: Performance Metrics of the Models

The comparative performance metrics of each model, providing a clear visualization of their effectiveness is illustrated in Fig. 7. Table II: Summary of House Price Prediction Trends Using Deep Learning. Table I. hyper parameter used for prediction.

TABLE II. SUMMARY OF REVIEWED HOUSE PRICE TRENDS USING DEEP LEARNING TECHNIQUES FOR HOUSE PRICE PREDICTION

MODEL	MSE	MAE	R^2 Score
NeuTs	0.0007	0.0214	0.9849
TimeGPT	0.0018	0.0317	0.9960
TCN	0.0026	0.0084	0.9975
CNN-LSTM	0.0173	0.0106	0.9995
DeepAR	0.0033	0.0178	0.9968

NeuTS demonstrated exceptional performance with an MAE of 0.0214 and an MSE of 0.0007, indicating its proficiency in capturing temporal dependencies. TimeGPT also performed well, achieving an MAE of 0.0317 and an MSE of 0.0018, showcasing its strength in long-term forecasting. TCN delivered robust results with an MAE of 0.0084 and an MSE of 0.0026, reflecting its effectiveness in handling sequential data. CNN-LSTM model with an MAE of 0.0106 and an MSE of 0.0173, making it suitable for high-dimensional time series data. DeepAR provided probabilistic forecasts with an MAE of 0.0178 and an MSE of 0.0033, enhancing interpretability and reliability.

The visualizations in Fig. 8, Fig. 9, Fig. 10 are created using Plotly, and Fig. 11, developed with Streamlit Application, illustrate the price forecasts for New York, NY.

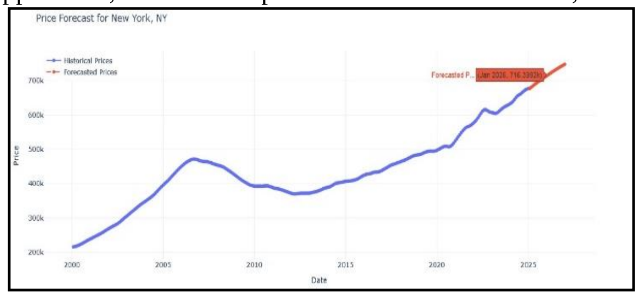

Fig. 8: Price Forecast using NeuTS model

Forecasted house prices in New York, NY, showing historical trends and future projections are depicted in Fig. 8.

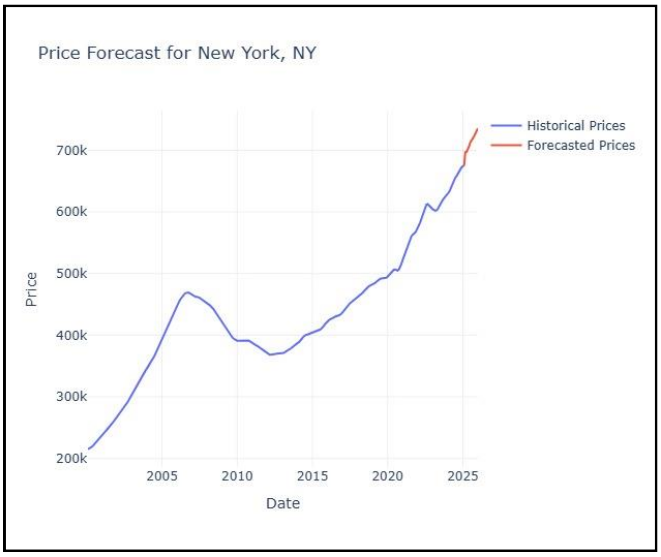

Fig. 9: Price Forecast using TCN model

Fig. 9 showcases the forecasting capabilities of the TCN model, providing a detailed comparison between historical prices and forecasted values.

Fig. 10: Price Forecast using TimeGPT model

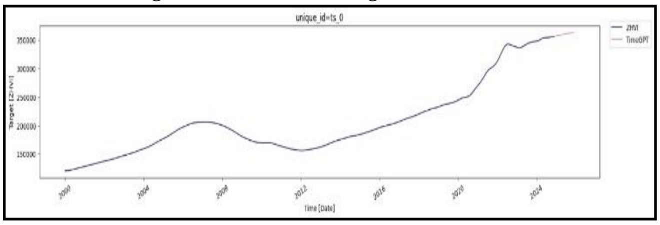

Fig. 10: Price Forecast using TimeGPT model

Historical and forecasted house prices using the TimeGPT model, showing trend consistency over time in Fig. 10.

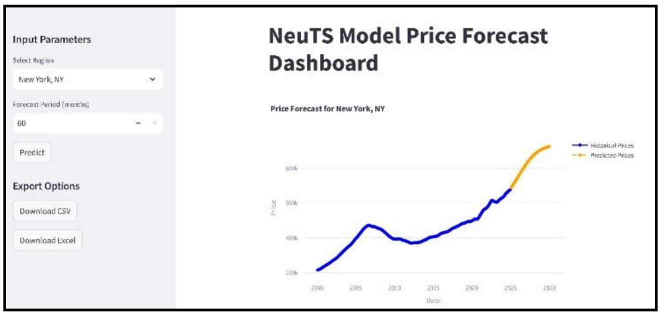

Fig. 11: Price Forecast's Frontend (using NeuTS model) implemented via Streamlit Application

Fig. 11, on the other hand, offers an interactive interface for stakeholders to explore different scenarios and observe the model's predictions in real-time. These visualizations help stakeholders analyze housing market trends and make informed property investment decisions.

V. DISCUSSION AND FUTURE CHALLENGES

The fact that one has to know is that Deep learning models are often sensitive to hyperparameters such as learning rate, batch size, and network architecture. Extensive tuning and experimentation are required to find optimal settings. Complex models with many parameters can overfit to the training data, especially if the dataset is not large enough. Regularization techniques and cross-validation are essential to mitigate this issue.

While metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and the R-squared (R^2) score are commonly used, their restricted scope can hinder a comprehensive evaluation, potentially leading to an incomplete understanding of model performance. This underscores the need for more diverse and robust metrics to provide a holistic assessment across various applications and data characteristics.

To enhance the evaluation of real estate house price prediction models and address the challenge of limited reliable metrics, we propose several future enhancements tailored to this domain. These include developing specialized metrics that capture the unique aspects of real estate valuation, such as location-specific trends and property features. Implementing multi-metric evaluation frameworks will provide a comprehensive assessment of model performance, considering factors like accuracy, robustness, and market adaptability. Establishing benchmarks specific to different property types and regions will enable more meaningful comparisons. Additionally, advanced visualization techniques, such as interactive maps and trend analyses, will improve interpretability for stakeholders. Continuous monitoring systems with feedback loops will ensure models remain accurate and relevant as market conditions evolve. By adopting these approaches, we aim to provide more robust and practical evaluations, ultimately enhancing the utility of house price prediction models in the dynamic real estate market.

VI. CONCLUSION

This study evaluates five deep learning models for real estate price prediction using the Zillow dataset [20]. NeuTS achieves the highest predictive accuracy with the lowest MSE (0.0004), making it the most reliable for price forecasting. TCN demonstrates the best balance, with the lowest MAE (0.0084), a competitive MSE (0.0026), and the highest R^2 Score (0.9975), ensuring both accuracy and robustness. These findings underscore the effectiveness of deep learning in real estate price prediction and offer valuable insights for model selection.

REFERENCES

[1] S. Mehtab and J. Sen, "Stock Price Prediction Using CNN and LSTM-Based Deep Learning Models," *2020 International Conference on Decision Aid Sciences and Application (DASA)*, Sakheer, Bahrain, 2020, https://ieeexplore.ieee.org/document/9317207

[2] Y. Piao, A. Chen, and Z. Shang, "Housing Price Prediction Based on CNN," 2019 *9th International Conference on Information Science and Technology (ICIST), Hulunbuir, China, 2019*,https://ieeexplore.ieee.org/document/8836731

[3] Garza, C. Challu, and M. Mergenthaler-Canseco, TimeGPT-1. 2024, https://arxiv.org/abs/2310.03589

[4] Azul, José Morales, Max Mergenthaler, Nikhil Gupta, TimeGPT-1. 2024,https://github.com/Nixtla/nixtla

[5] Haixu Wu, Jiehui Xu, Jianmin Wang, Mingsheng Long, "DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks," NeurIPS 2021 Conference Proceedings, 2021, https://github.com/thuml/TimesNet.

[6] S. Shi, X. Qiu, Y. Ru, and X. Tan, "A DeepAR-Based Neural Network for Time Series Forecasting," in *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, Tianjin, China, 2023,https://doi.org/10.1109/DOCS60977.2023.10294820

[7] W. Dai, Y. An, and W. Long, "Price change prediction of ultra high frequency financial data based on temporal convolutional network," *Procedia Computer Science*, vol. 199, pp. 1177-1183, 2022,https://doi.org/10.1016/j.procs.2022.01.149

[8] Jia Heping, Guo Yuchen, Zhang Xiaobin, Liu Dunnan, Yang Zhenglin, Zheng Yaxian, Feng Shuhai, Zeng Dan, Yao Xun. "A Price Prediction Method Based on Wavelet Packet Decomposition and TCN-LSTM Combination," Chinese Patent CN117952646A, filed January 9, 2024, and published April 30, 2024,https://patents.google.com/patent/CN117952646A/en.

[9] D. Yao, G. Cong, C. Zhang, X. Meng, R. Duan, and J. Bi, "A linear time approach to computing time series similarity based on deep metric learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 10, pp. 4554-4571, Oct. 1, 2022 , https://doi.org/10.1109/TKDE.2020.3047070

[10] Di Yao, NeuTS,2020, http://github.com/yaodi833/NeuTS

[11] Bryan Lim, Sercan Ö. Arık, Nicolas Loeff , Tomas Pfister, Temporal Fusion Transformers for interpretable multi-horizon time series forecasting ,International Journal of Forecasting ,Volume 37, Issue 4,2020, https://arxiv.org/abs/1912.09363

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/CVPR.2016.90

[13] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks. *NeurIPS Conference Proceedings*. https://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[14] Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ICML Conference Proceedings*. https://arxiv.org/abs/1502.03167

[15] Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *ICLR Conference Proceedings*. https://arxiv.org/abs/1412.6980

[16] A. Varma, A. Sarma, S. Doshi and R. Nair, "House Price Prediction Using Machine Learning and Neural Networks," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 2018, pp. 1936-1939, https://doi.org/10.1109/ICICCT.2018.8473231

[17] Shi, S. (2023). Comparison of Real Estate Price Prediction Based on LSTM and LGBM. *Highlights in Science, Engineering and Technology*, 49, 294-301. https://doi.org/10.54097/hset.v49i.8521

[18] Shi, S., Qiu, X., Ru, Y., & Tan, X. (2023). A DeepAR-Based Neural Network for Time Series Forecasting. *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, Tianjin, China. https://doi.org/10.1109/DOCS60977.2023.10294820

[19] Dai, W., An, Y., & Long, W. (2022). Price change prediction of ultra high frequency financial data based on temporal convolutional network. *Procedia Computer Science*, 199, 1177-1183. https://doi.org/10.1016/j.procs.2022.01.149

[20] R. Barton, L. Spencer, Zillow, "Zillow Dataset", 2006 ,[https://www.zillow.com/research/data/.