Predicting the S&P/Case-Shiller U.S. National Home Price Index

Likang Wang
Doctor of Philosophy
Department of Computer Science and Engineering
Hong Kong Univeristy of Science and Technology
Hong Kong
lwangcg@connect.ust.hk

Abstract

Predicting future trends in housing prices is a task of considerable economic and social value. In this paper, we first conduct a comprehensive and in-depth review of existing methods for forecasting housing price indices. Subsequently, we design several approaches specifically for forecasting the future values of the S&P/Case-Shiller U.S. National Home Price Index, achieving notably successful outcomes. An in-depth analysis of our experimental results is provided, along with suggestions for further development directions.

1 Introduction

In light of the topic, this task focuses on predicting the future values of the S&P/Case-Shiller U.S. National Home Price Index based on historical data provided by the Federal Reserve Economic Data (FRED). Predicting this index is crucial for economic analysts and investors, as it offers in-depth insights into the real estate market and serves as a critical indicator of economic trends. An examination of related academic literature and projects on Kaggle and GitHub reveals a variety of methods encompassing statistics, econometrics, machine learning, and deep learning, all aiming to leverage economic, financial, and social data to enhance the accuracy and reliability of predictions. This paper begins by outlining the current solutions in the field, categorizing and comparing them comprehensively. Then, it specifically formulates several methods designed for forecasting housing price indices, followed by extensive experimentation and detailed analysis.

2 Background

The rapid growth of the US real estate market over the past decade underscores the importance of forecasting the future trends of the S&P/Case-Shiller U.S. National Home Price Index. Available data from the Federal Reserve Economic Database (FRED) introduces multi-disciplinary methods for prediction, igniting interest among researchers, economists, and analysts. We think that existing methods can be categorized into the following four types based on their operational principles.

2.1 Time Series Methods

Time series analysis employs historical trends, cyclicality, and seasonal variations to forecast the future. Traditional methods include ARIMA (Box et al., 1976), Exponential Smoothing (Brown,

Preprint. Under review.

1959), and Seasonal Decomposition (Cleveland et al., 1990), exemplary for their capability to capture long-term dependencies and seasonal changes, though they may struggle with nonlinear relationships.

- Exponential Smoothing assigns exponentially decreasing weights to past observations, focusing particularly on more recent data. For instance, the Holt-Winters (Winters, 1960) method, designed for forecasting seasonal sales, incorporates seasonal components into the prediction model.
- Seasonal Decomposition models, like STL decomposition (Cleveland et al., 1990), dissect time series into trend, seasonality, and random components. Decomposing the S&P/Case-Shiller index's seasonal fluctuations clarifies long-term trends and adjusts for seasonality.

2.2 Machine Learning Models

Models such as Linear Regression (Draper and Smith, 1998) and Ridge Regression (Hoerl and Kennard, 1970) learn from extensive historical data to accurately manage relationships among multiple variables, excelling when data showcase linear relationships.

- Linear Regression assumes a linear relationship between input features and the output, offering rapid, interpretable results for datasets with approximately linear connections among variables.
- Ridge Regression introduces an L^2 regularization term to Linear Regression to manage multicollinearity, optimizing performance, especially when predictor variables outnumber observations.

2.3 Deep Learning Models

Models like LSTM (Hochreiter and Schmidhuber, 1997) and Transformers (Vaswani et al., 2017) excel in capturing patterns within high-dimensional and long sequential data, ideally suited for tasks involving long-term dependencies and complex dynamics.

- LSTM, specifically designed to manage long-term dependencies, counters the vanishing gradient problem common in traditional RNNs (Rumelhart et al., 1986), making it suitable for analyzing prolonged time associations in price prediction tasks.
- Transformers leverage self-attention mechanisms allowing the model to directly connect any two points in a sequence, facilitating rapid, flexible global learning without sequentially passing information like RNNs.

2.4 Econometric Models

Econometric models merge economic theory with empirical testing to deeply interpret quantitative relationships within economic behaviors.

- VAR models (Sims, 1980) anticipate future values by considering the interactions among variables, proving particularly effective in analyzing complex market dynamics across multiple variable time series.

2.5 Summary of Existing Methods

As shown in Table 1, different methods each have their strengths and weaknesses in predicting the S&P/Case-Shiller U.S. National Home Price Index. The choice of the most suitable method is contingent upon the specific demands of the prediction task, the type and quantity of data available, the need for model interpretability, and the desired level of forecast accuracy. Therefore, we will further explore in the following sections which methods are best suited to this data.

3 Methodology

As discussed previously, the selection of models is highly dependent on the characteristics of the data itself. Therefore, we begin by analyzing the raw data of the S&P/Case-Shiller U.S. National Home Price Index.

Table 1: Comparative Analysis of Existing Mainstream Home Prcice Index Forecasting Approaches

Method Category	**Advantages**	**Disadvantages**	**Application Example**
Time Series Methods	Excels in capturing seasonality and trends	Struggles with nonlinear relationships	Holt-Winters, STL decomposition
Machine Learning Models	Strong in learning relationships among multiple variables	Requires large data, Linear assumptions may not always hold	Linear Regression, Ridge Regression
Deep Learning Models	Efficient in handling long time series and complex dynamics	High computational cost, Lacks interpretability	LSTM, Transformers
Econometric Models	Strong interpretability, Solid theoretical foundation	Linear assumptions, Low adaptability to new market dynamics	VAR model

3.1 Data Preprocessing

The time range of the raw data spans from January 1, 1975, to February 1, 2024. However, data from January 1, 1975, to December 1, 1986, are missing. Consequently, we selected the home price index from January 1, 1987, to February 1, 2024, as our effective data. The visualization of this data is displayed in Figure 1.

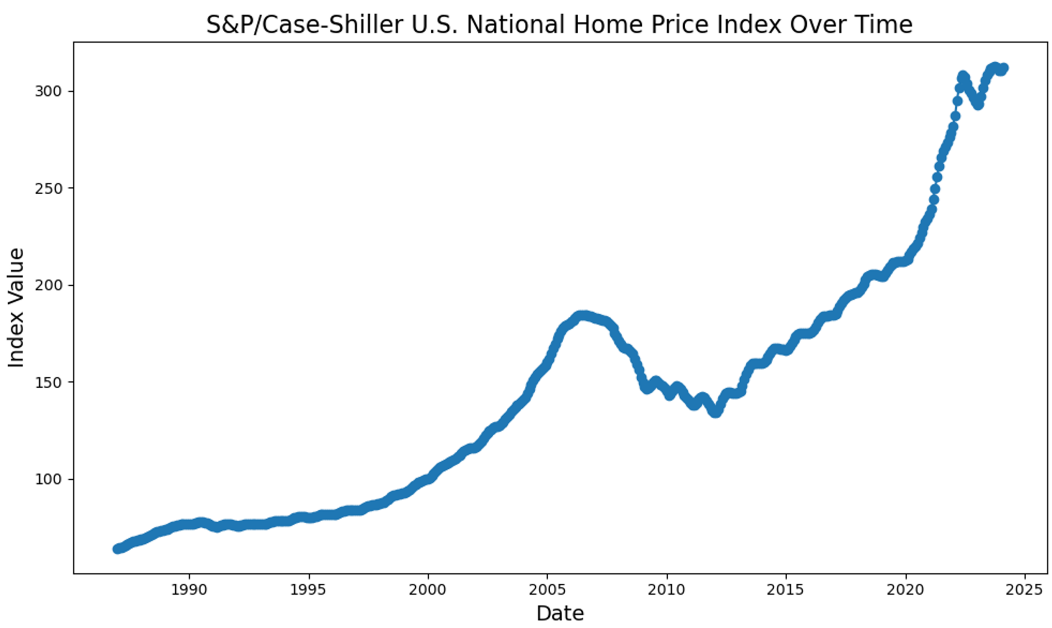

Figure 1: Changes in the U.S. National Home Price Index Over Time.

3.2 Data Division

Given the continuity and relevancy of the home price index over time, it is imperative to maintain the relative sequence of the data. Therefore, we divided it into two consecutive time intervals. The earlier interval was used for training, and the latter for validation. Given the lack of a specified train-test split strategy in the task description, we adhered to the most common partitioning settings. Specifically, we designated the first 90% of the data as the training set and the subsequent 10% (approximately the last five years) as the validation set. After initially assessing a model's performance under this partitioning, we will further investigated its behavior with less and more training data to explore its limits.

3.3 Performance Metrics

Since the model's inputs and outputs are continuous floating-point numbers, we believe that a combination of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Relative Error (MRE) can comprehensively evaluate the model's performance. These metrics are calculated as shown in Equations 1, 2, and 3, where n is the total number of time frames, $\hat{y}_i$ is the predicted home price index at time i, and y_i is the actual home price index. Both RMSE and MAE measure absolute error in housing prices; the key difference is that the former is more sensitive to larger errors, while the latter does not overlook small errors. MRE is a normalized metric, less affected by the scale of the data. Additionally, as home prices undergo drastic changes over time, the same absolute error can have different implications at different times. The MRE can account for this issue.

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

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{2}$$

$$\text{MRE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{3}$$

3.4 Model Selection

Given that the data set contains only 446 valid entries, to prevent overfitting, we opted to start with models of lower fitting capacity.

3.4.1 Low-capacity Models

As the quintessential machine learning model, we first attempted fitting using Linear Regression. As depicted in Figure 2, under the default 9:1 data split, the model could only roughly summarize the overall upward trend of the housing price index without capturing its finer details. Table 2 further corroborates this point, revealing the model's poor performance on both the training and validation sets.

To distinguish whether poor model performance was a result of an excessively small training set, we expanded the training data. Yet, as shown in Figure 3 and Table 3, even when training with 99% of the data—an extreme case—the performance remained inadequate. This implies the insufficiency of a univariate linear regression's capacity for this task.

Table 2: Numerical Analysis of Linear Regression Results Trained on the First 90% of Data.

Metric	**Value**
Root Mean Squared Error (Validation)	74.118
Mean Absolute Error (Validation)	69.303
Mean Relative Error (Validation)	0.2398
Root Mean Squared Error (Train)	16.325
Mean Absolute Error (Train)	12.945
Mean Relative Error (Train)	0.1036

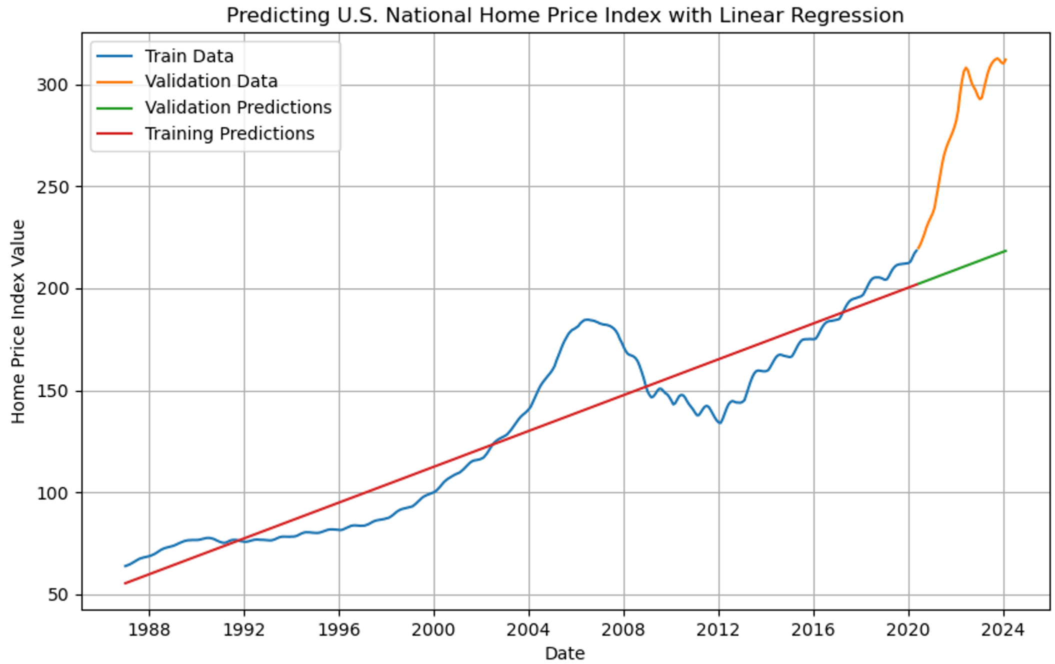

Figure 2: Visualization of Linear Regression Results Trained on the First 90% of Data.

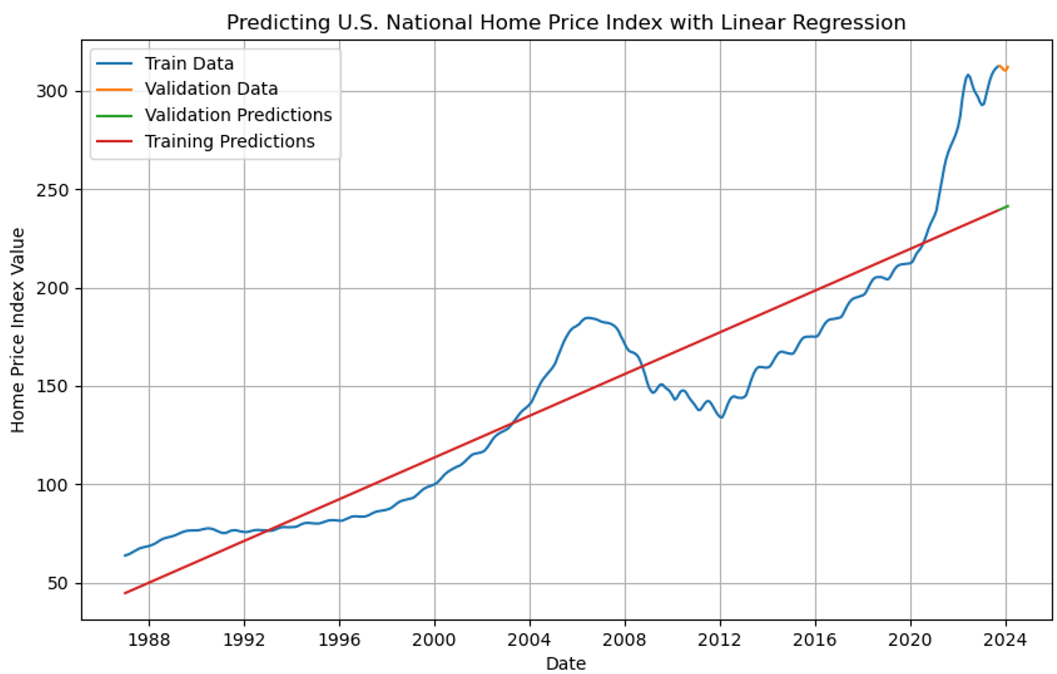

Figure 3: Visualization of Linear Regression Results Trained on the First 99% of Data.

3.4.2 High-capacity Models

To achieve a more reliable fit, we could either introduce more input variables or switch to models with greater fitting capacities. Given the numerous models with higher fitting capacities than linear regression, we initially aimed to increase the model's capacity. Our first choice was LSTM because it effectively utilizes temporal correlations and requires less data than models such as Transformers.

Table 3: Numerical Analysis of Linear Regression Results Trained on the First 99% of Data.

Metric	**Value**
Root Mean Squared Error (Validation)	71.062
Mean Absolute Error (Validation)	71.050
Mean Relative Error (Validation)	0.2280
Root Mean Squared Error (Train)	23.945
Mean Absolute Error (Train)	18.719
Mean Relative Error (Train)	0.1334

Model Configurations Considering the limited amount of original data, we decided to control the number of hidden layers, setting the count to 2, with each layer having 48 neurons. Following common neural network practices, we used the Adam (Kingma and Ba, 2014) optimizer with a learning rate of 0.01, performing 5,000 gradient updates. To enhance training stability, the original data was linearly normalized by its maximum and minimum values during training. The model we implemented in PyTorch (Paszke et al., 2019) could be trained and validated in just 10 seconds on a standard personal-use laptop equipped with a Ryzen 7945HX and RTX 4060.

Result Analysis As illustrated in Figure 4, the predicted housing price index almost perfectly matched the actual values, suggesting highly effective model performance. The quantitative results presented in Table 4 further validate this conclusion, indicating the model's fitting capability was appropriate.

In order to push the model to its limits, we substantially reduced the proportion of the training set to 60%, thereby extending the validation period to approximately 15 years. As shown in Figure 5 and Table 5, the model's performance, while slightly degraded compared to the default setting, still maintained sufficient accuracy.

Table 4: Numerical Analysis of LSTM Results Trained on the First 90% of Data.

Metric	**Value**
Root Mean Squared Error (Validation)	0.9513
Mean Absolute Error (Validation)	0.7600
Mean Relative Error (Validation)	0.0017
Root Mean Squared Error (Train)	0.2978
Mean Absolute Error (Train)	0.2191
Mean Relative Error (Train)	0.0012

Deliverability Assessment:Considering that the performance metrics of the aforementioned model meet the requirements of the task and its accuracy is sufficient to provide useful guidance for real estate investment, we believe it can serve as the final deliverable for this project. Of course, actual business needs may be more complex, and different application scenarios might require distinct criteria for the model. For instance, if we require the model to be more reliable for long-term forecasts, exploring Sequence-to-Sequence models like the Informer (Zhou et al., 2021)might be more appropriate; if we need the model to possess stronger interpretability, then trying traditional machine learning methods such as Random Forest Regression (Breiman, 2001) might be more suitable.

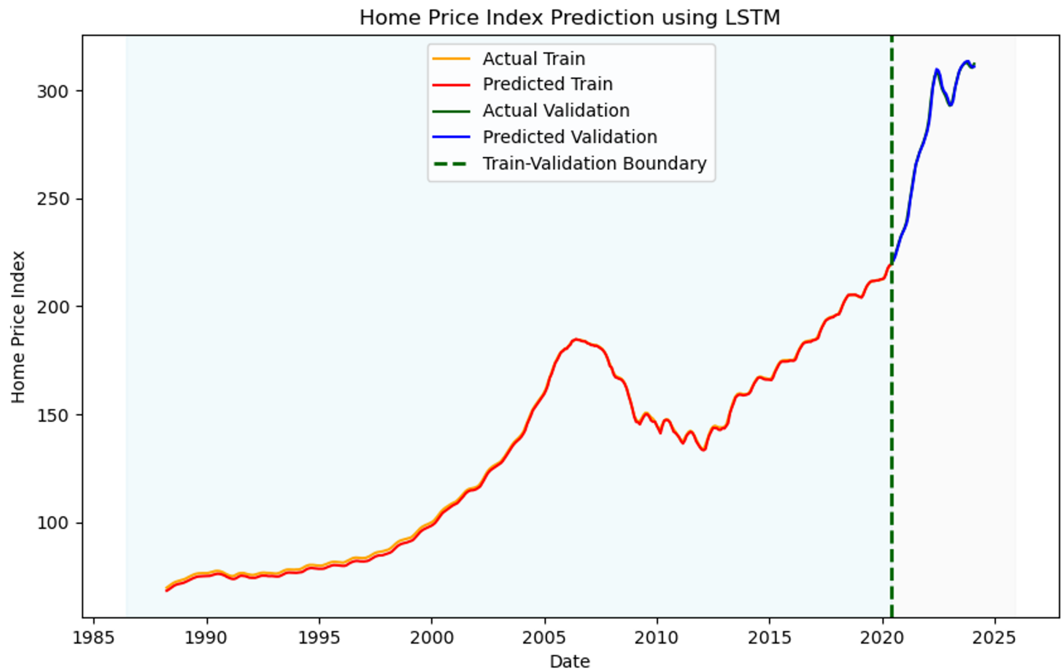

Figure 4: Visualization of LSTM Results Trained on the First 90% of Data.

Table 5: Numerical Analysis of LSTM Results Trained on the First 60% of Data.

Metric	**Value**
Root Mean Squared Error (Validation)	1.2827
Mean Absolute Error (Validation)	1.0668
Mean Relative Error (Validation)	0.0042
Root Mean Squared Error (Train)	0.2124
Mean Absolute Error (Train)	0.1696
Mean Relative Error (Train)	0.0008

4 Conclusion

This paper systematically analyzes the strengths and weaknesses of existing methods for forecasting the U.S. housing price index and presents a viable solution based on an analysis of the data's characteristics. Our comprehensive and extensive experiments confirm that this solution excels in both effectiveness and efficiency, meeting the needs of both the project and practical applications. Moreover, we offer valuable insights into potential improvements for its use in various real-world applications.

References

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 1976.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Robert Goodell Brown. Statistical forecasting for inventory control. 1959.

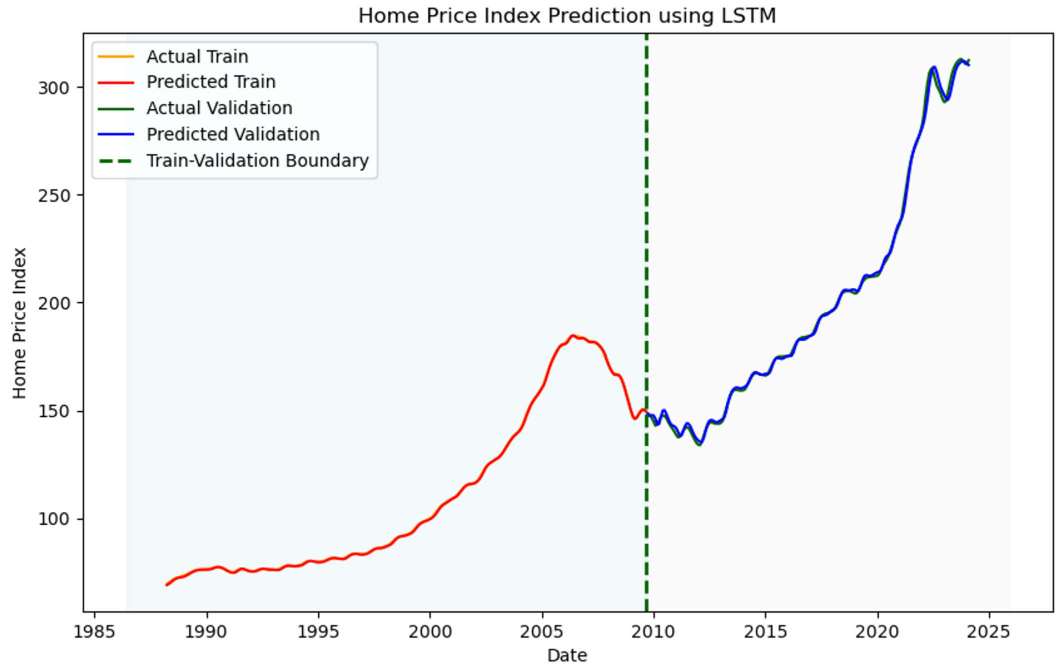

Figure 5: Visualization of LSTM Results Trained on the First 60% of Data.

Robert B Cleveland, William S Cleveland, Jean E McRae, Irma Terpenning, et al. Stl: A seasonal-trend decomposition. *J. Off. Stat*, 6(1):3–73, 1990.

Norman R. Draper and Harry Smith. *Applied Regression Analysis*. John Wiley & Sons, 3rd edition, 1998.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library, 2019.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

Christopher A Sims. Macroeconomics and reality. *Econometrica: Journal of the Econometric Society*, pages 1–48, 1980.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Peter R Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342, 1960.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11106–11115, 2021.