Team 51: Affordable Housing Analytics

Yuanshan (Tracy) Hu
Georgia Institute of Technology
Atlanta, Georgia
yhu437@gatech.edu

Nicholas Kousen
Georgia Institute of Technology
Atlanta, Georgia
nkousen3@gatech.edu

Manoj Mohanan Nair
Georgia Institute of Technology
Atlanta, Georgia
mmn9@gatech.edu

Santhanu Venugopal Sunitha
Georgia Institute of Technology
Atlanta, Georgia
ssunitha3@gatech.edu

Grant Windes
Georgia Institute of Technology
Atlanta, Georgia
gwindes3@gatech.edu

1. Introduction and Motivation

Homes are the biggest purchases people make in their lifetime. However, consumers often have difficulty understanding current market trends and how much they can afford. Housing data compiled by the Census Bureau and individual listings provided by real estate applications lacks intuitive visualizations and recommendations based on affordability or pricing projections. These resources are suboptimal since potential buyers cannot focus on neighborhoods based on their own affordability. Furthermore, the lack of future valuation predictions means buyers have little insights into the returns on investment (ROI). Our model aims to make regional trends and affordability projections more accessible and applicable by addressing these shortcomings.

2. Problem Definition

1. Develop statistical models to predict housing prices to provide personalized affordability recommendations based on net household income.
2. Enable informed, data-driven decision making with intuitive visualizations of dynamic choropleth maps, charts, and graphs for affordability and regional pricing predictions.

3. Literature Survey

Traditional affordability measurements uses the Mortgage Bankers Association (MBA) index, which compares mortgage applications to home sales for forecasting analysis. However, Duca suggests that MBA performance may falter during periods of uncertainty [1], while Ben-Shahar et al propose also considering fossil fuels, resource consumption, education level, etc. when assessing affordability [2]. Ganong et al concluded that additional factors affecting affordability included wage gaps, supply constraints, and migration trends, through their regional income analysis of metropolitan areas regarding affordability [3]. Finally, McCabe's studies concluded that community engagement also correlates with residential stability, providing useful insights on neighborhood valuation, even though it can be difficult to quantify [4].

By studying the 2008 housing collapse, Schelkle provides insights into "double triggers" or the combination of unemployment and negative equity that leads to mortgage defaults and foreclosures [5]. Alm et al focused on foreclosure rates and the housing collapse to understand how it influenced tax revenue and budget allocations for local governments, which profiles additional users who could benefit from an affordability model [6]. And as unemployment rises, epidemic patterns of housing prices emerge that provide interpretations of the broader economy [7].

Macroeconomics can also be influenced by demolition and controlled construction, affecting supply and demand, which can contribute to volatility in forecasts [8]. Adding perspectives on cycles and bubbles can help understand differences in motivations and affordability of investors versus buyers to improve model accuracy as well [9]. These insights offer perspectives to be incorporated in our model for holistic affordability analysis.

Basic housing price predictions can use linear models, and Lowrance's study of more than four decades worth of housing data provides useful guidances on regularization and feature selection [10]. But hedonic regressions can also be used to estimate housing prices based on attributes, such as home-type and design. Finally, "Days On Market" models examine housing prices beyond intrinsic evaluation of attributes, which can support well-rounded valuation to help potential home buyers [11].

Limsombunchai suggested Artificial Neural Networks (ANN) machine learning models to evaluate housing prices, which offer greater accuracy and provide foundations for later improvements in deep learning for estimation techniques [12]. Sarip et al compared ANN to fuzzy least-squares regression-based (FLSR) models to evaluate accuracy and complexity, finding it capable of capturing functional relationships of dependent and independent variables [13]. Time series forecasting can alternatively be used to model median house listing prices, and Siami-Namini et al concluded that Long-Short-Term Memory (LSTM) preformed the best due to the back propagation optimization approach in deep neural networks [14]. Yu et al also approached affordability predictions using LSTM models to predict housing prices, and detailed its effectiveness over time series predictions and convolution neural networks [15]. As such, we believe a univariate LSTM model would best support a comprehensive affordability model.

4. Proposed Method

4.1 Intuition

Our approach includes the development of personalized affordability metrics and house pricing predictions through an interactive user interface (UI), providing insights into the financial risks of home ownership. Prospective home buyers will find our tool complements existing tools, such as Zillow and Redfin, in their home search process. By focusing our tool on macro (state) and micro (county) levels of the housing market, prospective home buyers will be able to analyze whether a specific listing is within their affordability price range, as well as detect if the home is an outlier within a county's median pricing.

4.2 Description of Our Approaches

Data Extraction and Cleaning

The data was procured from Zillow [16] and includes US zip codes with corresponding housing median list prices and foreclosure rates from 2010 to present. The entire dataset includes 2.57M+ data points. The zip codes were aggregated at the county and the state levels to compensate for null values and create more comprehensive datasets. Since housing prices were populated horizontally, the price columns were transposed vertically using Python Pandas and indexed based on time, which is more compatible with the LSTM neural network.

Price Prediction Model

To predict housing prices over the next six months, we applied LSTM deep neural networks on median listing prices across US counties and states for the past eight years. We implemented LSTM learner using the Keras library to model time series price data and predict prices six months into the future. The moving forward window for LSTM networks, which is the size of inputs used to predict the next data points, was varied between two and ten months with one hidden layer initially selected. The training data was normalized using Scikit-learn's Min-Max scaler by removing the mean and scaling the price variance. This is common in machine learning estimations since non-uniform data can impede model performance.

Using the findings from our initial experiments, we tuned hyperparameters of the model using GridSearchCV and Keras Wrapper to find optimal epochs, batches, and optimizers. The optimized model was used to predict the next six months of housing prices for all 50 states, storing our results as a .csv file for visualization in our web application. Due to time constraints, we limited price projections to the state level.

Algorithms and Additional Data Analyses

Affordability calculations utilize individual's household net income to calculate the number of years one's full salary would be needed to pay for a home.

$$Affordability = \frac{Median\ Home\ Price}{Net\ Household\ Income}$$

Visualizing affordability presents a new metric to prospective buyers that is otherwise often overlooked when calculating the true cost of home ownership.

Bollinger bands support affordability by utilizing median listings prices to indicate bottom (oversold) or top (overbought) assets using a simple moving average (SMA) and standard deviation of housing prices to indicate a buyer's versus seller's market. Traditional bollinger bands utilized in stock trading use a 20-day or 21-day SMA. This is derived from a typical year having around 251 trading days. Dividing the 251 trading days by 12 months equals 20.9 trading days each month, which gives a trader a rolling monthly average for assets to determine if the bottom or top of a bollinger band is a valid signal to buy or sell. We utilized 20-month SMAs since our data is calculated monthly, whereas stocks are daily.

$$MiddleBand = 20\ month\ Simple\ Moving\ Average$$

$$UpperBand = 20\ month\ SMA + (20\ month\ std\ price * 2)$$

$$LowerBand = 20\ month\ SMA - (20\ month\ std\ price * 2)$$

User Interface and Design Criteria

Users input their salary to receive personalized affordability metrics, mortgage magnitude (ratio of median price ÷ income), and choropleth maps utilizing D3.js and Python to highlight affordable neighborhoods. Our model forecasts future costs, provides current market state insights, and visualizes true costs of homeownership from holistic perspectives personalized to budget constraints. Our UI also includes price predictions from LSTM learner, transformed bollinger bands, and foreclosure data, as well as home affordability calculations. Additionally, a date slider is included to enable exploration of median prices and foreclosure data over time for trend analysis.

When designing our interface visualization, we followed standard on-screen guidelines, such as arranging the charts and maps in a grid, making use of whitespace, and reducing clutter. Additionally, we followed heuristic principles for effective interface design, specifically using principles of consistency, simplicity, and feedback, by incorporating design elements that follow platform conventions. This ensures that our interface helps users ascend the usability learning curve as rapidly with as little experience as possible, creating an invisible interface. Our interface updates automatically in response to user actions, such as changing the net income values or hovering a mouse cursor over the choropleth map to view underlying tooltips, providing feedback that is immediate and informative. The colors in the map also refresh showing the changes to the backend algorithms as users adjust timeframes and / or income inputs. Overall, through the design of our interface, we aimed to narrow the gulfs of execution and evaluation that users may experience from feedback cycles.

Figure 1: Interface Dashboard Featuring Median House Listing Price by State

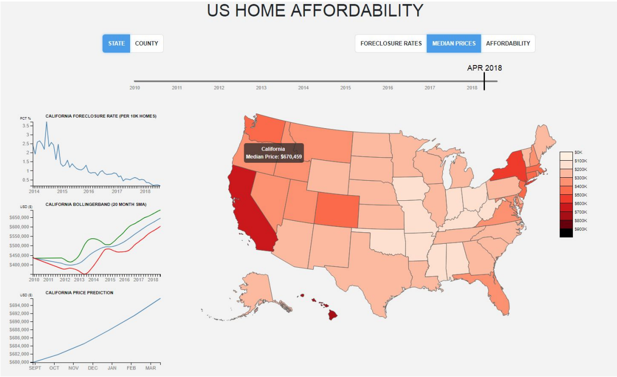
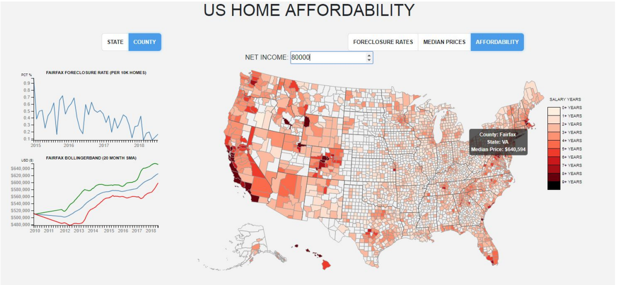


Figure 2: Interface Dashboard Featuring Affordability by County



NET INCOME: 80000

5. Experiments and Evaluation

5.1 Description of Testbed

Prediction Model

1. What are the optimal hyperparameters to reduce prediction error?
2. Can the model reliably predict median home prices?
3. Can the model predict fluctuations or up / down trends in housing prices?

Visualization

1. Does our tool positively influence home purchasing behavior?
2. Does our tool utilize user-centered design principles, including standard formatting and frequent feedback for effective interface interactions?

5.2 Experiments and Observations

Price Prediction Model

We used root-mean-squared error (RMSE) and mean-squared error (MSE) to measure the accuracy of our predictions for initial model tuning. RMSE is highly sensitive to large errors and helps identify when predictions have high deviation from actual pricings. Our goal is to minimize RMSE in conjunction with MSE to obtain the best model for predicting Zillow house prices.

Experiment Design

Experiment 1

We implemented an LSTM network with one hidden layer of 32 units, batch size of five, and trained the model over moving forward window between three to ten months for 500 epochs. Preliminary results are from the Manhattan (New York County, NY) housing price data from 2010, split 75:25 into training and test sets. Predicted prices were tested against the test dataset, and the RMSE and MSE were plotted for observation.

Figure 3: RMSE with 500 Epochs

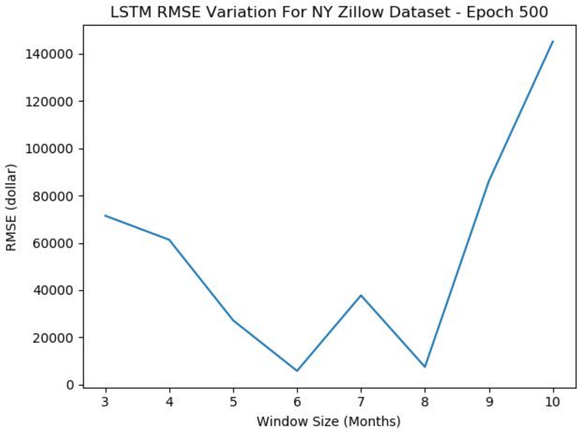
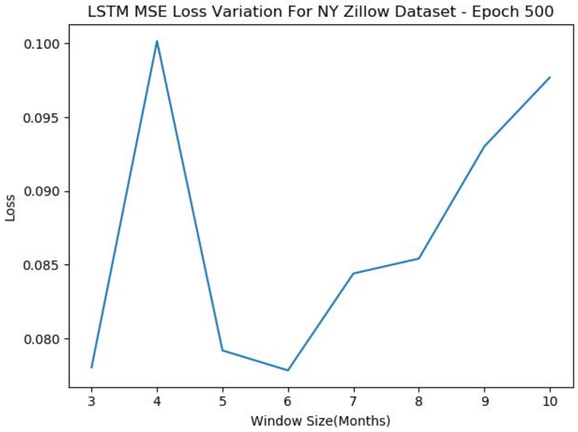


Figure 4: MSE with 500 Epochs

(Normalized Values)



Minimum RMSE occurs at six months with notable decreases between five to eight months, indicating this is the optimal window size. Smaller windows (fewer data points) could result in overfitting, while larger windows are prone to underfitting.

Experiment 2

We changed epoch size from 500 to 1,000 to evaluate the effect of epochs on accuracy.

Figure 5: RMSE with 1,000 Epochs

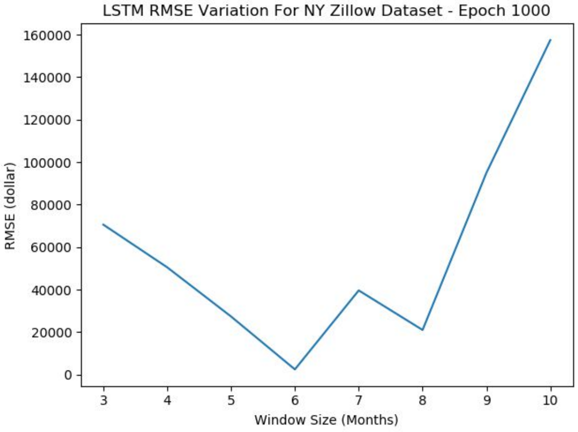
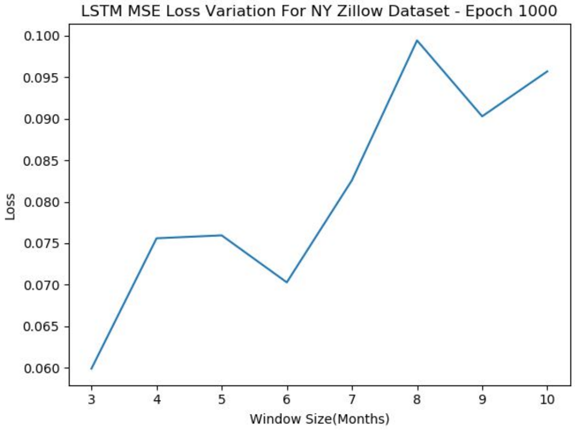


Figure 6: MSE with 1,000 Epochs

(Normalized Values)



Again, minimum RMSE can be observed at six months. The minimum MSE also improved compared to *Experiment 1*. As expected, larger epochs improved accuracy due to additional iterations of training using. However, too many epochs could result in overfitting.

Experiment 3

Second hidden layer with 64 units was added to the model to evaluate effects on performance. All other parameters were held constant at 1,000 epochs.

Figure 7: RMSE with Hidden Layer

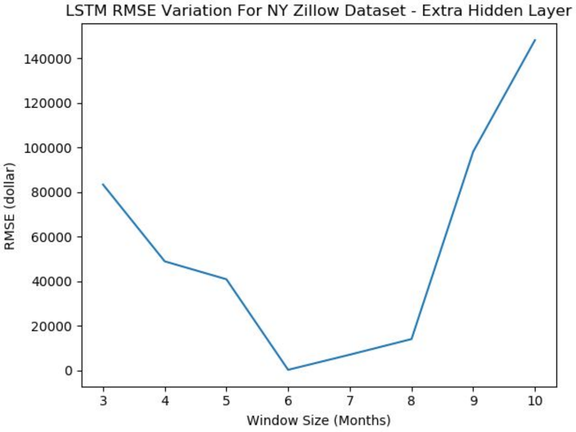
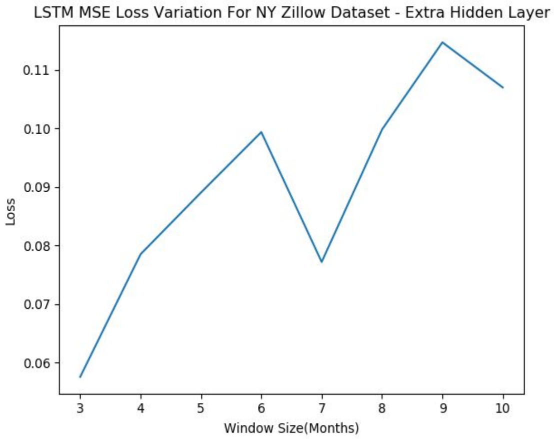


Figure 8: MSE with Hidden Layer (Normalized Values)



The minimum RMSE occurs at six months; MSE graph shows increasing error at window sizes greater than seven, confirming window size six or seven months as optimal.

Experiment 4

Utilizing the previous parameters as starting points, we tested the following hyperparameters using the full Manhattan dataset spanning from January 2010 through August 2018, split 80:20 into training and test sets. The LSTM was created with two hidden layers of 64 neuron each. The parameters below were evaluated using GridsearchCV over a range of window sizes (2, 3, 4, 5, 6).

Table 1 : Additional Parameters Tested

Hyperparameter	Range
Epochs	{1000,1500,2000,2500}
Batch Size	{5,10,15}
Optimizer	{‘rmsprop’,’adam’}

Optimal epoch size remained consistent with previous experiment at size 1,000.

Batch size represent the number samples that is utilized for training by the network at once. Lowering the batch size can reduce memory overhead and enable faster training. However, there is a tradeoff between speed and accuracy since lower batch sizes cause gradient estimates to be less accurate. After hyperparameter tuning, we observed that a batch size of 15 led to the lowest RMSE.

We evaluated two stochastic gradient descent algorithms, Adaptive Moment Estimation (Adam) and Root Mean Square Propagation (RMSProp), and Adam performed better due to faster convergence and lower bias.

Optimal Hyperparameter - {'batch_size': 15, 'epochs': 1000, 'optimizer': 'adam'}

Figure 9 : RMSE with Optimal Parameters

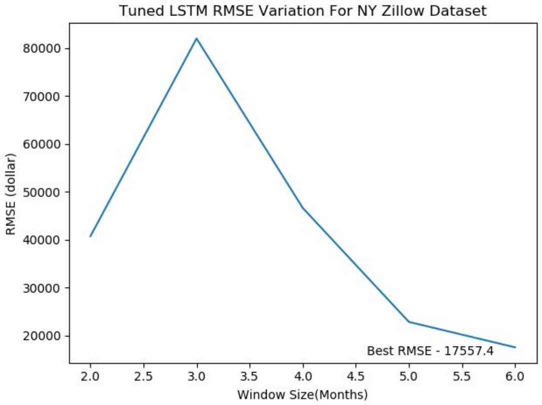
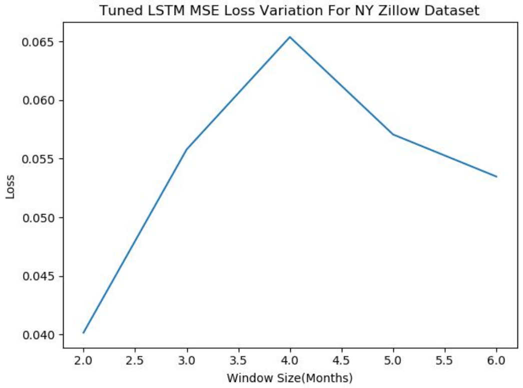


Figure 10 : MSE with Optimal Parameters (Normalized Values)



After the hyperparameter tuning, we observed that the loss (MSE) error for window size = 6 went up, while window size = 2 was the new minimum RMSE (17,557.4). This can be attributed to improved learning from the increased batch size (15 compared to 5 in experiments 1-3) due to more observations per timestep.

Experiment 5

Since the current model converged relatively quickly, we increased the number of neurons from 64 to 256 for hidden layers and observed a further decrease in RMSE. Again, the minimum RMSE (2,948.13) is observed at the 2-month window.

Figure 11 : RMSE with 256 Neurons

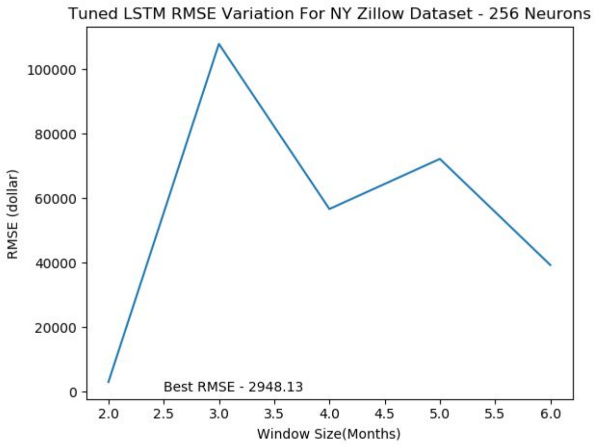
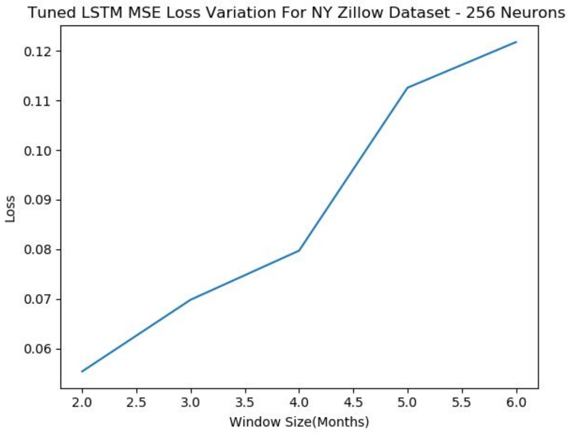


Figure 12 : MSE with 256 Neurons (Normalized Values)



Experiment 6

To avoid overfitting due to increased number of neurons, we regularized the network by increasing the drop out values.

Dropout is a regularization technique used to reduce the neural network complexity, thereby avoiding overfitting. This hyperparameter determines the proportion of units to be dropped from the layer during the linear transformation of samples, reducing interdependency between the units in the layers and helps with better generalization. The lowest RMSE was obtained with a dropout of 0.2 and 0.5 for the two hidden layers respectively. Again, the 2-month window showed the lowest error, with the minimum RMSE at 2,584.5.

Figure 13 : RMSE with Dropouts

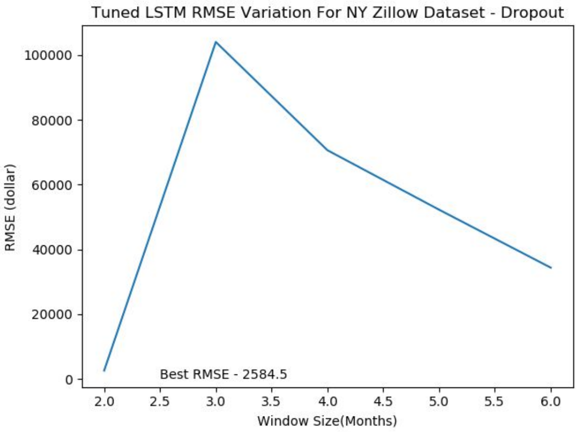
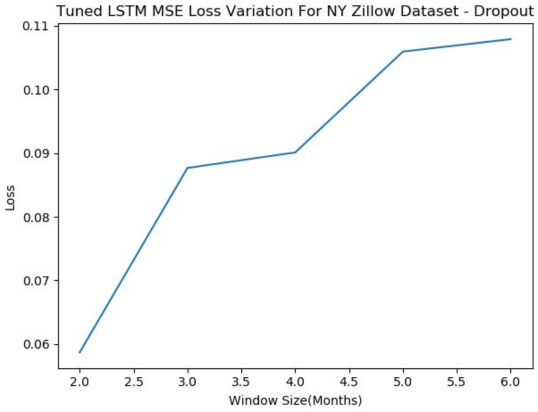


Figure 14 : MSE with Dropouts (Normalized Values)



Experiment 7

To further reduce the RMSE, we created a bidirectional LSTM by implementing a second LSTM learner, which takes inputs in reverse chronological order. Bidirectional LSTM is known to improve the LSTM performance by training a second LSTM network alongside the first, wherein data is fed in a reversed order to the second network. This helps eliminate ambiguity and the learner gets better context of the sample space. Overall, further reductions in error was observed using the Bidirectional LSTM, with the best RMSE (178.53) occuring at the 2-month window.

Figure 15 : RMSE with Bi-directional LSTM

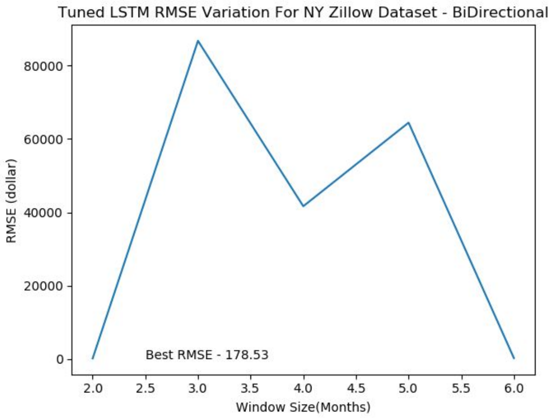
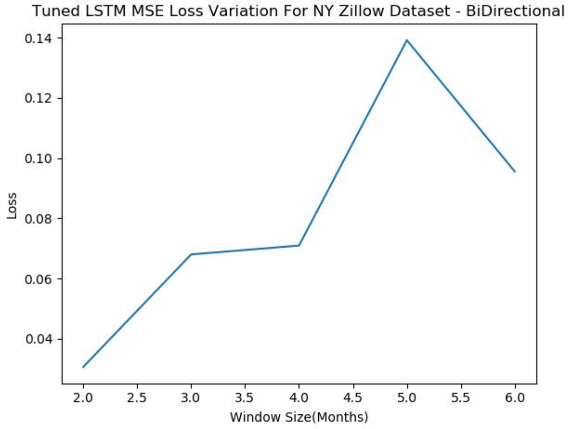


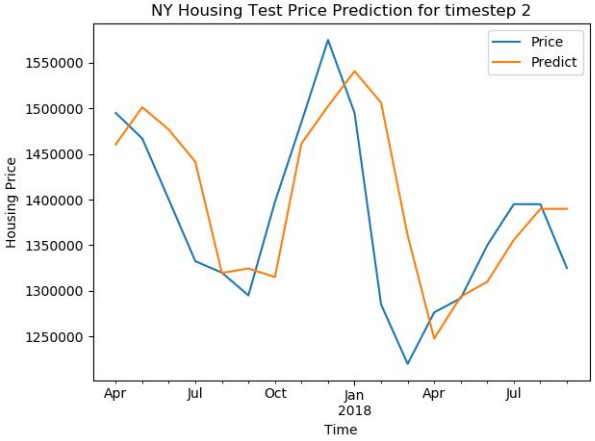
Figure 16 : MSE with Bi-directional LSTM (Normalized Values)



Experiment 8

We plotted price predictions against the Manhattan test data over an 18-month period below. The predictions followed actual price trends, were sensitive to fluctuations in housing prices, and were able to accurately predict up and down trends. We backtested predicted values to available pricing data and obtained the mean-absolute error (MAE) of $52,579 with an average percent error of 4.2%. Using the same methodology, we trained models and applied to the full dataset for all 50 states.

Figure 17: Final Price Predictions with Tuned Bidirectional LSTM model



5.3 Visualization

To evaluate the effectiveness of the interface design, user studies were conducted by each of the team members to ensure the ease of use and understanding. Among the elements evaluated included font sizes and chart labels, which were adjusted based on user feedback. Arrangement of the charts, map, and buttons were also relocated through iterative designs to ensure proper visibility of the interface, models, and visualizations with minimal scrolling.

6. Conclusion and Discussions

6.1 Conclusion

Our tool presents an intuitive approach at helping home buyers target affordable areas for purchase and maximizing their ROI. Our visualizations support proactive analysis of housing affordability dynamically by regions of the US with personalized recommendations based on user input data. Our visualizations of foreclosure rates, bollinger bands, and short-term price predictions provide valuable information on market trends to help lower debt to equity ratio and mitigate market risks.

While our tool can be improved on by including additional metrics that influence home prices, such as school district ratings or local crime statistics, it can nevertheless serve as a valuable feature for integration with or use alongside existing real-estate search sites, educating users on financial feasibilities of home ownership and positively influencing home purchasing behavior.

6.2 Future Work

One of the challenges when using the existing Zillow dataset was the missing data for certain counties and zip codes.

Given additional time, we would validate, clean, and aggregate data from additional sources to cover more zip codes and counties and create comprehensive views of the state and county level pricing and foreclosure data.

We would also use mortgage interest rates, unemployment, median income, school district ratings, crime statistics, or market sentiment analysis to create a multivariate learner to improve accuracy over our univariate learner. Expanding price predictions to detect long-term trends would also make the tool more useful for long term investors. This would include adding user inputs for mortgage rates from banking APIs and expected down payments, which would further personalize the model’s resulting affordability calculations.

Finally, development of a mobile friendly user interface that is responsive to different screen sizes and touch input would improve the user experience and broaden the potential user base. Additional user studies with interviews and surveys of a broader audience would also provide useful feedback to further iterate our interface through the design life cycle, creating a more effective user experience.

6.3 Distribution of Team Effort

All team members have contributed similar amount of effort.

7. References

[1] Duca, J. V. (1996). Can mortgage applications help predict home sales? *Economic Review - Federal Reserve Bank of Dallas*.

[2] Ben-Shahar, D., & Warszawski, J. (2016). Inequality in housing affordability: Measurement and estimation. *Urban Studies, 53*(6), 1178-1202.

[3] Ganong, P., & Shoag, D. (2017). Why Has Regional Income Convergence in the U.S. Declined? *Journal of Urban Economics, 102*, 76-90.

[4] McCabe, B. J. (2013). Are Homeowners Better Citizens? Homeownership and Community Participation in the United States. *Social Forces,91(3),* 929-954.

[5] Schelkle, T. (2018). Mortgage Default during the U.S. Mortgage Crisis. *Journal of Money, Credit and Banking, 50(6),* 1101-1137.

[6] Alm, & Leguizamon. (2018). The housing crisis, foreclosures, and local tax revenues. *Regional Science and Urban Economics, 70*, 300-311.

[7] Fogli, A., Hill, E., & Perri, F. (2012). The Geography of the Great Recession. *Seminar on Macroeconomics, University of Chicago Press, 9(1),* 305-311.

[8] Gurran, N., & Bramley, G. (2017). *Urban Planning and the Housing Market International Perspectives for Policy and Practice.*

[9] Diappi, L. (2013). *Emergent Phenomena in Housing Markets Gentrification, Housing Search, Polarization.*

[10] Lowrance, R.E. (2015). Predicting the Market Value of Single-Family Residential Real Estate.

[11] Zhu, H., Xiong, H., Tang, F., Liu, Q., Ge, Y., Chen, E., & Fu, Y. (2016). Days on market: Measuring liquidity in real estate markets. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 393-402.

[12] Limsombunchai, V. (2004). House Price Prediction: Hedonic Price Model vs. Artificial Neural Network. *IDEAS Working Paper Series from RePEc,* IDEAS Working Paper Series from RePEc, 2004.

[13] Sarip, A. G., Hafez, M. B., & Daud, Md. N. (2016). Application of fuzzy regression model for real estate price prediction. *Malaysian Journal of Computer Science, 29(1),* 15-27.

[14] Siami-Namini, S. and Namin, A. (2018). *Forecasting Economics and Financial Time Series: ARIMA vs. LSTM.*

[15] Yu, L.A., Jiao, C., Xin, H., Wang, Y., & Wang, K. (2018). Prediction on Housing Price Based on Deep Learning.

[16] Zillow, 2018 Retrieved from https://www.zillow.com/howto/api/APIOverview.htm