COMPARATIVE ANALYSIS OF MACHINE LEARNIG AND DEEP LEARNING MODELS FOR REAL ESTATE PRICE PREDICTION IN DUBAI

ANEESA KP

FINAL THESIS REPORT

MAY 2025

DEDICATION

I dedicate this work to my dear husband, for always believing in me, encouraging me, and standing by my side through every step of this journey. Your love and support have been my greatest strength.

To my family, thank you for your constant support, patience, and motivation.

A special thanks to my thesis supervisor, Dr. Saravanapriya, for guiding me throughout this program, always keeping me on the right track, and motivating me even during moments of doubt. Your support and timely communication meant a lot.

I would also like to thank Dr. Shubra Goyal for being so cooperative, kind, and always ready to help.

ACKNOWLEDGEMENT

I would like to express our sincere gratitude to Liverpool John Moores University, UK, in collaboration with UpGrad, for their invaluable support during our research journey. The university's rich resources, expert faculty, and access to advanced facilities greatly contributed to the quality and success of this study. I am truly thankful to everyone at the university who played a part in making this work possible.

ABSTRACT

Dubai's rapid urbanization, luxury-driven infrastructure, and status as a global investment hub have positioned its real estate market as one of the most dynamic and competitive in the world. Accurate real estate price prediction in Dubai is essential due to the market's volatility, high-value transactions, and the diverse factors influencing property prices—including location, property type, developer reputation, and global economic conditions. Price prediction helps investors, brokers, developers, and policymakers make informed decisions, minimize financial risk, and identify profitable opportunities in a market driven by both local and international demand. However, predicting real estate prices in Dubai presents several challenges: the heterogeneity of properties, fluctuations due to economic cycles, regulatory changes, and limited availability of clean, structured data. To address these complexities, this study applies a wide range of machine learning and deep learning models that are well-suited for capturing non-linear relationships and learning from high-dimensional datasets. We evaluate the predictive performance of machine learning models such as Support Vector Regression (SVR), Random Forest (RF), Gradient Boosting Regression (GBR), Extreme Gradient Boosting (XGBoost), Ridge, and Lasso. These are selected for their strong baseline performance, interpretability, and robustness against overfitting. In parallel, advanced deep learning models—such as Deep & Cross Networks (DCN), Transformer, TabTransformer, Feature Token Transformer (FT-Transformer), and Gated Recurrent Units (GRU)—are used to explore their ability to learn deep feature interactions and temporal dynamics from complex datasets. Evaluation metrics including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2) are used to assess model accuracy and generalization. This comparative analysis aims to identify the most effective modelling techniques to enhance price prediction in Dubai's evolving real estate landscape.

Keywords: ML, DL, SVR, RF, GBR, XGBoost, Ridge and Lasso, DCN, FT-Transformer, TabTransformer, Transformer, GRU.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ML	Machine Learning
DL	Deep Learning
SVR	Support Vector Regression
RF	Random Forest
GBR	Gradient Boosting Regression
DCN	Deep and Cross Network
FT-Transformer	Feature Token-Transformer
GRU	Gated Recurrent Unit
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error

CHAPTER 1

INTRODUCTION

1.1 Background of the study

Dubai is famous for its fast-paced lifestyle, luxury cars, modern architecture, Burj khalifah etc. These factors made Dubai into the center stage of the world in terms of civilization and to be named to be one of the fastest growing cities in the world. Although success is always seen as the tip of the iceberg, we often do not focus on the solid fundamentals laid down by the historical leaders who were the backbone of building such an iconic city in a few decades. As a rapidly developing city, Dubai continues to attract a growing number of investors and residents. The city's robust financial and economic growth has positioned its real estate market as a major attraction, drawing interest from around the world. Because of tax exemption, high capital appreciation, high rental yields (which range from 6% to 8% in many cases), and the availability of affordable housing, Dubai has become a highly attractive destination for both local and international property investors, offering a lucrative market with opportunities for strong returns on investment and long-term growth. Following the COVID pandemic, Dubai's real estate market has experienced significant growth, with price prediction becoming increasingly important. However, predicting prices remain challenging due to the market's frequent fluctuations and dynamic nature.

Additionally, Dubai's strategic location at the crossroads of Europe, Asia, and Africa plays a crucial role in its success. The city serves as a global business hub, benefiting from an open economy, free trade zones, and world-class infrastructure, such as the Dubai International Financial Centre (DIFC) and Jebel Ali Port, the largest port in the Middle East. This has attracted multinational corporations and entrepreneurs alike, further boosting the demand for commercial and residential real estate.

Machine learning models are designed to learn patterns from data and make predictions or decisions without being explicitly programmed. These models can be applied across various fields, including real estate, healthcare, finance, marketing, and more. By selecting the appropriate model and enhancing its performance through data preprocessing and hyperparameter tuning, businesses can achieve greater accuracy and efficiency across a wide range of applications

(Elnaeem Balila and Shabri, 2024) conducted a comparative analysis of Dubai real estate price prediction using various machine learning models, including SVM, EEMD-SD-SVM, Gradient Boosting, Random Forest, KNN, ANN, and Linear Regression. Machine Learning The models were evaluated based on metrics such as R-squared, RMSE, MAPE, and MSE. The EEMD-SD-SVM model emerged as the best predictor, demonstrating a high R-squared value and lower MAPE, MSE, and RMSE values. Their analysis found out that the area size of the property is very important to check when a buyer wants to do the transaction. (Alshamsi, n.d.) has done a predictive analysis of Dubai real estate by using different ML models and found that random forest is the best model that gives higher R-squared value than others with findings that apartment size and number of bathrooms and bedrooms plays a vital role in predicting the price. (Alfalasi, n.d.) implemented various machine learning models, including the Generalized Linear Model, SVM, and Neural Networks, to predict real estate prices. The findings revealed that the Generalized Linear Model outperformed the other models in terms of prediction accuracy. (Mostofi et al., 2022) conducting real estate price prediction by using deep neural networks adopting principal component analysis (PCA). The PCA-DNN models outperform DNN and SRA-DNN models. The PCA-DNN model can manage the high-dimensional dataset. The adoption of the PCA-DNN model helps to offset the reduced prediction accuracy caused by the limited number of price records and the categorical nature of feature columns. This model proves valuable in real estate and construction applications, where the lack of medium and large datasets typically hampers price prediction accuracy. (2019 IEEE Symposium Series on Computational Intelligence (SSCI), 2019) has done a study on real estate price prediction eXtreme Gradient Boosting (XGBoost) on historical sales data with visual content. And the study successfully found that XGBoost can predict the price with high performance.

This study focuses on predicting the price of Dubai real estate using both Deep Learning (DL) and Machine Learning (ML) models. While extensive research has been conducted on Dubai's real estate price prediction using ML models, studies exploring the application of DL models in this context are limited. Predicting the price helps investors, real estate brokers, buyers, sellers, stakeholders to make decisions on investing in Dubai real estate with the help of current trends. We compare the performances of machine learning models such as support vector Regression(SVR), Random Forest(RF), Gradient Boosting Regression(GBR), Extreme Gradient Boosting (XGBoost), Ridge and Lasso with advanced deep learning models such as Transformer, Deep and Cross Networks (DCN), TabTransformer, Feature-Token Transformer

(FT- Transformer) and Gated Recurrent Units(GRU) with the performance matrices Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and R-Squared.

One of the key limitations of ML models is their reliance on manual feature selection, which can be time-consuming and may not always capture the most relevant patterns in the data. In contrast, DL models are capable of automatically handling large and complex datasets, such as those generated from real estate transactions, with greater efficiency. This ability allows DL models to scale more effectively, where traditional ML models often struggle. Adavanced DL models give higher accuracy by determining the deep patterns. These models have the scalability and robustness featuress to handle large and dynamic datasets like real estate datasets

Although ML models can solve complex problems, they may not always yield results as robust as those produced by DL models. As real estate transaction datasets continue to evolve and grow in complexity, it becomes increasingly important to uncover hidden patterns and relationships among the features that influence price prediction. Advanced DL models are particularly well-suited for this task, as they excel at complex pattern recognition and automatic feature selection.

1.2 Problem Statement

Extensive research has been conducted on real estate price prediction in Dubai using machine learning (ML) models. However, studies focusing on price prediction of Dubai Real Estate with deep learning (DL) models remain limited. Therefore, this analysis takes a broader, global perspective on real estate price prediction using DL models. In this study, focusing on both ML and DL models. Comparing real estate price predictions globally using deep learning gives much idea about how deep learning techniques are trying to find deep patterns with complex datasets like real estate than Machine Learning. During covid-19 times, the house prices were comparatively less because of lack of demand and travel restrictions. But by the end of 2022, there will be a very noticeable rise in price due to removing travel restriction. The study uses the Dubai house price data from Kaggle with 20 different columns to find trends and predict price using LSVM (Linear Support Vector Machine), Generalized Linear Regression and Neural Networks with findings of Generalized Linear 1 model is best performing model with correlation 71.1% (Alfalasi, n.d.). Similarly, the study of Dubai Apartment prices using different ML algorithms like SVR, Random Forest, Decision tree, Linear regression, K-nearest neighbor, boosted regression model on the Dubai real estate dataset from Kaggle of 38 features with the findings Random Forest performs well with R-squared value 73.12% (Alshamsi, n.d.). There is another study to predict the price of 1 BHK apartment by using different modern Models like SVM, EEMD-SD-SVM, Random Forest, KNN, GBM and ANN. With the help of these models, they could find the trend going on and the factors which strongly to drive the price predictions with the evaluation matrices that minimize the prediction errors. EESD-MD-SVM model has achieved this minimal prediction error with R2 value 0.541, MSE=0.090, RMSE=0.3010 and MAPE=3.3310 (Elnaeem Balila and Shabri, 2024). Using a variety of machine learning models, including ensemble learning algorithms based on boosting (Gradient Boosting Regressor, Extreme Gradient Boosting, and Light Gradient Boosting Machine) and bagging (random forest and extra-trees regressor), the study examined house price prediction in Alicante, Paris, and found that no algorithm outperformed the others with 94,024 records (Mora-Garcia et al., 2022). Another study of predicting house prices using Random Forest technique with Boston housing dataset with 14 features. The above study has come up with the fact that the housing prices are correlated with different factors like location, city, number of rooms, how old the property is etc. And the finding is the Random Forest model with R-squared value 90% (Adetunji et al., 2021). In the study of comparison analysis of ML and DL models against Hedonic regression models with the aim of reducing the human involvement in the price prediction to get better accuracy predictions using models. The study used different ML, DL and Hedonic models (ANN, KNN, BFPM, RF, SVM) and found out that the RF model outperformed against all other models with the R-squared value 86%, RMSE 0.21 and MAEP 1.14% (Yazdani, 2021).

Price prediction using deep learning (DL) offers improved accuracy by uncovering deep patterns and complex relationships between features within the dataset. In research involving apartment transaction data from Switzerland, various DL models were evaluated using Mean Squared Error (MSE) as the performance metric. Among them, the meta-model demonstrated the highest predictive accuracy, achieving an MSE of 0.0332 (Walthert et al., n.d.). The real estate price prediction system utilizing deep learning (DL) is based on 27,988 transaction records from Taiwan. Two frameworks were proposed to capture both spatial and temporal features: the Building Environment Encoding for Price prediction (BEEP), which employs a Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM) model, and the Location-Enhanced Price Prediction (LEPP) framework, which uses a shallow Recurrent Neural Network (RNN) (Chiu et al., 2022). The study of price prediction using DNN, PCA-DNN (Principal Component Analysis-Deep Neural Network) which compares the benchmark model as SRA. The study uses evaluation matrices such as MAE, MSE and MAPE and found out that PCA-DNN outperforms well with 98% prediction accuracy (Mostofi et al., 2022). Another study happened to predict the real estate indices of 6 stocks included in the sector Liquid45(LQ45) by using a bi-LSTM model. The study used matrices RMSE and MAPE and among the stocks BSDE, CTRA, PTPP, PWON, SMRA and WIKA, BSDE has highest accuracy value R-Squared value 98.86% (Hansun et al., 2022). These days price predictions using unstructured data like images also make sense. This study uses 1000 images from real estate portals to predict the price with the help of convolutional neural network which extracts information from visual features. They found out that when the sampling size increases the value of adj. R2 also increases (Despotovic et al., 2023). People are always attracted to visual or images. There is another study which uses image processing by using proposed method pairwise comparison method with the interior images of properties from At Home Co., Ltd. The proposed method achieved higher accuracy than other methods which is 68.8% (Wang et al., 2019). There is another study which uses CNN models to predict property prices with the help of 71,809 entries in which both master data (size, number of rooms/bathrooms, area etc.) and image data. The study concludes that combining master data and image data minimizes the prediction error and maximizes the accuracy with the value of RMSE 43,469 and MAE 37,337 (Kucklick et al., 2021). Another study applies Deep Reinforcement Learning (DRL) combined with Gramian Angular Field (GAF) and Long Short-Term Memory (LSTM) models for real estate trading, using publicly available Australian real estate data. The findings indicate that the proposed model, integrating DRL with GAF and LSTM, achieves high predictive accuracy (Zhao et al., 2022). Similarly, another study uses DL algorithms BPNN and CNN to predict housing price with the open transaction dataset provided by the ministry of Taiwan and found out that CNN (Convolution Neural Network) performs better than BPNN with higher R-squared value 94.5% and lower RMSE, MAE, MAPE and RMSLE (Zhan et al., 2020). The office price prediction study using a Neural Network (NN) model utilized data from the Chinese Real Estate Index System. The findings revealed that simple Neural Network models demonstrated stable performance, achieving an average Root Mean Square Error (RMSE) value of 1.45% (Xu and Zhang, 2024). Another study uses AI based ML techniques to predict the price and they have used ML techniques because the dataset is not large enough (Tekouabou et al., 2024). The study of real estate auction prediction employed Artificial Intelligence (AI) to develop a forecasting model. Genetic Algorithm (GA), Artificial Neural Network (ANN), and regression analysis models were applied to real estate auction cases of apartments in Seoul. The results showed that the Genetic Algorithm model outperformed the others, achieving a

Mean Absolute Percentage Error (MAPE) of 8.86 and a Root Mean Square Error (RMSE) of 0.006 (Kang et al., 2020).The real market price prediction (real estate index s&p500-60)using GRU,LSTM,CNN and the GRU model gives the best predict among them with R value 99% (Rimal et al., 2024).

From the above-mentioned studies, some studies prove ML models perform well in predicting real estate prices and some of them prove DL model perform well in predicting real estate prices. In this study, Dubai real estate price prediction, using advanced ML and DL algorithms like SVR, RF, GBR, XGBoost, Ridge and Lasso, Deep and Cross Networks (DCN), Transformer, TabTransformers, Feature-Token Transformer (FT-Transformer) and Gated Recurrent Units (GRU) that provides the ability to handle more complex dataset like real estate, can adapt various type of data and predict more accurately comparatively simple linear models.

1.3 Aim and Objectives

The main aim of this research is to conduct comparative analysis study of real estate price prediction using advanced ML and DL models. Advanced ML and DL are popular for accurate predictive results. These models can handle tabular data and effectively identify and select the most significant features to build prediction models.

The following objectives can be formulated from the aims mentioned above.

- To gather historical data from relevant resources.
- To preprocess the data by addressing missing values, outliers and applying normalization techniques. Handle missing data by either removing less significant values or inputting them with mean, median or mode values. Identify and manage outliers using box plot or scatter plot. For normalization, apply Z-score normalization (Standardization) to bring the features to a common scale.
- To evaluate different ML and advanced DL models to determine which performs best.
- To evaluate the performance of proposed approach based on model evaluation matrices like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R-Squared.

1.4. Significance of the study

Predicting real estate prices has become increasingly crucial in today's market. This study on Dubai real estate price prediction aims to assist international investors, buyers, sellers, and brokers in making informed decisions regarding real estate investments. By providing accurate price forecasts, the study also helps reduce the occurrence of price bubbles, which occur when property prices rise above their actual market value without fundamental factors supporting the increase. Therefore, accurate prediction is vital.

To achieve this, various machine learning and deep learning algorithms are proposed. Machine learning algorithms such as support vector Regression (SVR), Random Forest, Gradient Boosting Regression (GBR), Extreme Gradient Boosting (XGBoost), Ridge and Lasso are particularly effective for structured or tabular data like real estate datasets. Additionally, deep learning algorithms Transformer, Deep and Cross Networks (DCN), TabTransformer, Feature-Token Transformer (FT- Transformer) and Gated Recurrent Units (GRU) are used to uncover deep patterns and complex relationships within the data, further enhancing the accuracy of price predictions. Together, these advanced models provide a robust approach to predicting real estate prices in the dynamic Dubai market.

1.5. Scope of the study

The scope of the real estate price prediction study is extensive. It involves collecting historical data, preprocessing it, and applying various machine learning and deep learning algorithms to determine which model performs best. A comparative analysis between these models is essential to identify the most effective approach. Understanding current market trends is crucial, as it helps guide investment decisions, whether to buy, sell, or hold property. Property prices are influenced by numerous factors, such as location, number of rooms, developer reputation, and the type of project (off-plan or ready to move).

Dubai has emerged as a global investment hub, with real estate playing a significant role in driving investments. Monitoring market trends is key when investing in real estate. This study provides valuable insights for investors, brokers, and stakeholders, helping them make well-informed decisions in Dubai's dynamic real estate market.

1.6 Structure of the Study

The components of the study are follows.

Chapter 1 - Introduction: This section introduces the overview and importance of the topic and research objectives

Chapter 2 - Literature Review: This section reviews existing study of the research; key factors influencing the research and identifying the gaps in the research

Chapter 3 - Research Methodology: This section explains the research methods and models used with dataset description, sampling methods and model building

Chapter 4- Implementation and Analysis: The findings are discussed in relation to the research questions and existing literature. The chapter also explores the study's implications and acknowledges any limitations encountered

Chapter 5-. Result and Evaluation: The findings are discussed in relation to the research questions and existing literature. The chapter also explores the study's implications and acknowledges any limitations encountered.

Chapter 6- Conclusion and Recommendation: This chapter summarizes the key findings and highlights the major contributions of the study. It also offers practical recommendations based on the results

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The literature review for Dubai real estate price prediction examines a range of studies and methodologies that have been employed to forecast property prices in dynamic markets. The prediction of real estate prices has become a critical area of research due to its importance in guiding investment decisions, policymaking, and market analysis. Various machine learning (ML) and deep learning (DL) models have been explored in recent years to enhance the accuracy of predictions by identifying complex patterns within large datasets. Traditional statistical methods, such as linear regression, have been widely used in earlier studies, but the growing availability of big data has led to a shift towards more advanced techniques capable of handling the multifaceted nature of real estate markets.

In the context of Dubai, the real estate market is influenced by unique factors such as rapid urban development, global investment flows, regulatory changes, and shifting market trends. Researchers have increasingly focused on leveraging advanced ML and DL algorithms to predict property prices, considering these diverse influences. Studies have applied models such as decision trees, support vector Regression (SVR), random forests, and gradient boosting Regression (GBR), as well as deep learning architectures like artificial neural networks (ANNs), and recurrent neural networks (RNNs), to improve predictive performance.

This literature review will explore the evolution of these predictive models, highlighting their strengths and limitations, particularly in the context of Dubai's real estate sector. It will also examine how emerging technologies, such as transformers and ensemble learning techniques, are being utilized to enhance the precision and robustness of price predictions. By reviewing existing studies and methodologies, this section aims to provide a comprehensive understanding of the current state of research and the potential for future advancements in the field of real estate price forecasting.

2.3 Machine Learning in Real Estate Price Prediction

Machine learning (ML) has revolutionized real estate price prediction by offering data-driven, accurate, and efficient methods for analyzing complex datasets. In the real estate industry, predicting property prices involves numerous factors such as location, property size, amenities, economic conditions, and market trends. Traditional models often fail to capture the intricate relationships between these features. However, machine learning algorithms excel in this domain due to their ability to learn from historical data, identify hidden patterns, and make highly accurate predictions. A study highlights that foreign nationals own 43% of Dubai's residential property, with foreign investment growing by 20% ($23 billion) from 2020 to 2022. A surge in Russian investment followed the Ukraine invasion, with Russians purchasing $2.4 billion in existing and $3.9 billion in in-development properties. It suggests increasing pressure on the United Arab Emirates (UAE) from Anti-Money Laundering (AML) bodies and expanding the Organization for Economic Co-operation and Development's (OECD) Common Reporting Standard (CRS) to include real estate. Dubai's market remains opaque, facilitating tax evasion and money laundering, especially through its residency-by-investment scheme and lack of real estate reporting. The low rate of suspicious transaction reports (0.1%) from real estate agents raises concerns about inadequate regulation (Alstadsaeter et al., 2024). Another study examines how U.S. monetary policy impacts Dubai's real estate market, given the UAE's currency peg to the U.S. dollar. Using monthly transaction data from 2014 to Q1-2022, the study identifies two key channels: mortgage costs and exchange rates. Findings show a strong inverse relationship between U.S. interest rate changes and Dubai's property demand, with a 1% rate hike reducing transactions by 17%. The exchange rate channel appears more influential than mortgage costs, as a stronger dollar makes Dubai real estate more expensive for foreign buyers. While the International Monetary Fund (IMF) supports the exchange rate peg for stability, fiscal policy adjustments—such as reducing property registration fees or modifying loan-to-value (LTV) ratios—could help mitigate the effects of U.S. monetary tightening. The study acknowledges limitations, including the lack of granular data on property types and pricing, which could influence varied responses across market segments (Shoukry Rashad and Farghally, n.d.). A 50-year analysis (1972–2021) of Land Use and Land Cover (LULC) in the UAE using 72 Landsat images and Machine Learning classifiers—Classification and Regression Tree (CART), Support Vector Machine (SVM), and Random Forest (RF)—found RF most accurate (85.11%–98.4%). Analyzing 46,146 polygons across four LULC classes, results showed desert/mountain areas declined from 97% to 91%, while built-up areas and vegetation rose to nearly 6% and 2.85%, respectively. The study demonstrates the value of ML and geospatial tools for monitoring land changes and guiding sustainable planning (Sultan et al., 2024). However, urban public squares remain overlooked in Middle East and North Africa (MENA) urban planning, with greater emphasis on iconic smart buildings rather than livable, community-driven spaces. The paper highlights the vital role of the real estate sector in integrating public squares into Dubai's smart urbanism to enhance social interaction and sustainability. Public spaces are essential for community well-being, providing play areas for children, green spaces for residents, and fostering social cohesion. Real estate stakeholders must recognize that sustainable urban development is a necessity, not a luxury. Collaboration among urban planners, designers, and local governments is crucial to ensure urban squares contribute to vibrant, inclusive communities. The dominance of skyscrapers has eroded civic spaces, weakening social life. A strategic policy framework should be established to incorporate well-designed public squares, enhancing livability in both new and existing urban developments (Ezzeddine, 2024). The study of comparing the effectiveness of eight machine learning (ML) algorithms, (ensemble empirical mode decomposition (EEMD)–stochastic(S)+deterministic(D)–support vector machine) EEMD-SD-SVM, SVM, gradient boosting, random forest, KNN, linear regression, ANN (Artificial Neural Network), and decision trees—for predicting property prices in Dubai. Key metrics like R^2, MSE, RMSE, and MAPE assess model accuracy and error rates. EEMD-SD-SVM and SVM excel in precision, gradient boosting and random forests handle complex data well, KNN captures localized patterns, and ANN performs best with large datasets. The study highlights the importance of model tuning, feature selection, and data preprocessing to optimize prediction accuracy for Dubai's real estate market (Elnaeem Balila and Shabri, 2024). The study of exploring real estate opportunities by presenting a machine learning application aimed at identifying properties listed significantly below market value in real time. Four different ML models—ensembles of regression trees, k-nearest neighbors, support vector regression, and multi-layer perceptron—were evaluated for their effectiveness in price prediction. Among these, ensembles of regression trees achieved the lowest mean absolute error, outperforming the other models in accuracy. The study also recommends for future work that advanced deep learning models excel in feature extraction, contributing to enhanced predictive power and insight into price-related features (Baldominos et al., 2018). A comparable study on house price prediction utilized the Boston housing dataset, implementing the Random Forest model due to its powerful ensemble methodology, which combines multiple decision trees to tackle complex predictive challenges more effectively than individual trees. The study employed R^2, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) as evaluation metrics to measure model accuracy. Results demonstrated that the Random Forest model achieved a high level of predictive precision, with price predictions falling within a ±5% margin of the actual values, showcasing its reliability and effectiveness in forecasting real estate prices. This success highlights Random Forest’s potential as a robust choice for similar applications in the real estate sector (Adetunji et al., 2021). A recent study on house price prediction in Abu Dhabi, UAE, implemented machine learning models such as Support Vector Machine (SVM), Random Forest (RF), Decision Trees, and K-Nearest Neighbors (KNN) to assess their effectiveness in forecasting property prices. The study involved data preprocessing, including dataset cleaning, followed by hyperparameter tuning to enhance model performance and identify the optimal model parameters. To evaluate the performance and reliability of each model, the researchers employed metrics like R^2, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). Among the models tested, Random Forest emerged as the top performer, achieving a high R^2 score of 96%, indicating its strong predictive power in this context (Al Marzooqi and Redouane, 2024). Another work explores various machine learning techniques, including decision trees, Kohonen maps, neural networks, and correlation analysis, applied to real estate price segment classification. The dataset quality was assessed using correlation analysis, while decision trees and Kohonen maps were employed to predict the target variable. Neural networks were also tested, though they demonstrated limitations in this software environment. Results were summarized in a comparative table, indicating that most methods achieved relatively high accuracy, particularly when numerous parameters were included, except for neural networks, which showed reduced accuracy in the specific software used (Borodulin et al., 2024). Another study on real estate price prediction in a Spanish city employed both boosting algorithms (Gradient Boosting Regressor, Extreme Gradient Boosting, and Light Gradient Boosting Machine) and bagging algorithms (Random Forest and Extra-Trees Regressor). After undergoing various preprocessing steps, hyperparameter tuning, and cross-validation, the study found that the bagging models tended to overfit compared to the boosting models. Boosting algorithms, particularly with large datasets, proved to be highly effective, highlighting their relevance for ML projects in the business sector where training and optimization are critical. Among all models, Gradient Boosting Regressor achieved the highest performance (Mora-Garcia et al., 2022). A study focused on minimizing human involvement in real estate price prediction compared traditional hedonic pricing (HP) methods with advanced machine learning (ML) and deep learning (DL) models, including Random Forest (RF), K-Nearest Neighbor (KNN), and Artificial Neural Networks (ANN). The results highlighted that HP methods, while valuable, are limited in capturing non-linear relationships between property features and prices. Conversely, RF and ANN models demonstrated the highest R-squared values along with superior MAPE and RMSE metrics, indicating strong predictive accuracy. ML and DL approaches excel in learning directly from data without requiring explicit function assumptions between input and output, allowing them to dynamically adjust their structures through parameter and hyperparameter tuning, thus enhancing predictive flexibility and accuracy (Yazdani, 2021). The article introduces a model-agnostic methodology for creating property price indices, aiming to incorporate non-linear and non-parametric models, such as machine learning (ML), into price index calculations. The primary innovation lies in leveraging individual out-of-time prediction errors as indicators of price change. Using a dataset of 29,998 commercial real estate transactions in New York from 2000 to 2019, the study finds that ML models achieve higher prediction accuracy than traditional linear models. However, ML models demonstrate greater sensitivity to the calibration data, leading to less stable results with smaller samples and potential estimation bias. To address this, strategies for reducing bias and balancing the bias-variance trade-off are recommended (Calainho et al., 2022). Another study aims to analyze house price trends and key pricing factors in Dubai's real estate market. Using the CRISP-DM framework, we identify drivers of property prices across different neighborhoods and evaluate various predictive models to select the best one for price forecasting. SPSS Statistics and Modeler were utilized for data preparation and statistical analysis, enabling a deeper understanding of dataset insights. Through comparative model analysis, we identified the Generalized Linear Model (GLM) as the top-performing model for accurate predictions. This framework can guide researchers and developers in applying a similar approach for property price forecasting in other regions (Alfalasi, n.d.). Another study proposes a novel approach to automated property valuation by integrating a convolutional neural network (CNN) with a fully connected neural network (MLP) to process numeric, categorical, and geographical data, achieving significantly improved accuracy. Tested on residential sales data from Greater Sydney, Australia, the fused model (Model 2) achieved a mean absolute percentage error (MAPE) of 8.71%, outperforming the baseline MLP-only model (Model 1) with a MAPE of 11.59%. The inclusion of geographical features via CNNs reduces spatial information loss and addresses omitted location and neighborhood characteristics. Spatially stratified splitting was employed to mitigate spatial autocorrelation, ensuring a robust and generalizable model (Lee et al., 2024). Another study introduces a two-stage machine learning approach to market segmentation for spatial econometric modeling, demonstrated with real estate data from three major Spanish cities. In the first stage, classification trees identify non-linear and non-orthogonal submarket boundaries, which are then represented as regional indicators. These indicators are incorporated into a second-stage hedonic price model that includes spatial interaction effects and covariates. The results show that this method improves model fit, predictive accuracy, and reduces spatial lag parameters by accounting for regional heterogeneity. While effective for short-term predictions, further research is needed to explore the temporal stability of market segments, considering seasonal effects and long-term trends such as gentrification. The above study emphasizes reproducibility by providing publicly available data and code, supporting validation and further investigation (Rey-Blanco et al., 2024). Another study develops a machine learning model to predict apartment price trends in Dubai using a Kaggle dataset of 1,905 apartments with 38 features, including numeric, categorical, and binary indicators, collected from real estate websites in 2020. Various machine learning techniques, such as Random Forest, Decision Tree, and Boosted Regression, were evaluated using cross-validation, with Random Forest identified as the most accurate model. Key determinants of prices included geographical coordinates, apartment size, and neighborhood characteristics, with size having a greater impact than the number of bathrooms. Features like concierge services, private pools, and water views also raised median prices (Alshamsi, n.d.). There is a paper reviews AI-based machine learning (ML) methods in real estate prediction, highlighting their growing use for forecasting property values and market trends. While traditional ML methods dominate due to limited data quality and availability, the study emphasizes ML's potential to enhance prediction accuracy and decision-making in areas like property valuation and construction costs. Challenges include improving data access, model explainability, and performance. The paper advocates open data initiatives and offers insights to guide future research, aiming to bridge gaps and advance intelligent real estate applications (Tekouabou et al., 2024). A study compares regression-based and machine learning-based models for rent price prediction, focusing on spatial dependence and large datasets. The regression approach employs the nearest neighbor Gaussian processes (NNGP) model, enabling kriging for large data. Machine learning methods include XGBoost, random forest (RF), and deep neural networks (DNN). Using Japanese apartment rent data with varying sample sizes, results show XGBoost consistently achieves the highest out-of-sample prediction accuracy across all sizes and error measures. Adding spatial coordinates to explanatory variables suffices for accounting for spatial dependence in RF. The findings highlight the superiority of machine learning models, particularly XGBoost, over regression methods for predictive accuracy in rent price modeling (Yoshida et al., 2022). Another study compares five machine learning (ML) algorithms—LR, SVR, KNN, XGBoost, and LSTM—with financial benchmarks (HES, TBATS, ARIMA) for predicting the prices of 30 REITs, stocks, and bonds across the US, UK, and Australia. ML models outperformed benchmarks, with SVR delivering the highest risk-adjusted returns. ML-based portfolios yielded three times higher returns and half the risk of traditional methods. KNN, SVR, and XGBoost showed the best performance in terms of RMSE, return, risk, and Sharpe ratio. The study suggests incorporating REITs for better diversification and portfolio performance, with future work exploring additional features and algorithms (Habbab and Kampouridis, 2024) . Machine learning (ML) plays a vital role in enhancing the efficiency, sustainability, and livability of smart housing within sustainable urban planning. This study demonstrates that ML can optimize energy consumption, waste management, and public safety, with notable results such as a 20% reduction in total energy use, a 15% increase in renewable energy consumption, and a 25% improvement in waste management efficiency. Additionally, public safety response times improved by 30%, and ML models achieved 92% accuracy in forecasting power use, traffic patterns, and air quality, reducing carbon emissions by 10%. The research highlights the importance of integrating ML into smart housing to drive sustainable urban development and improve resource management, energy efficiency, and overall quality of life in smart cities (Arabasy et al., 2024) . Investment in real estate is a popular choice, offering significant returns, with housing price trends reflecting current economic conditions. Multiple factors such as the number of bedrooms, locality, and proximity to amenities like roads, educational institutions, and malls influence housing prices. A study focuses on Pune as a case study to build a model for predicting real-time house prices across different localities using data from realtor websites like 99acres.com, magicbricks.com, and nobroker.com. Key features include area, bedrooms, and bathrooms, and regression techniques such as Multiple Linear Regression (OLS), Lasso, and XGBoost are used to compare accuracy and identify the best model. The above study also explores the effectiveness of machine learning (ML) and deep learning (DL) techniques, including random forest and gradient boosting, in improving prediction accuracy. A hybrid model combining residential property data from multiple sources aims to enhance predictions for both purchase prices and rental income, offering financial recommendations for potential investors (Suresh Yalgudkar and V. Dharwadkar, 2022). However, another study implies that Building Information Modeling (BIM) is transforming real estate valuation by enhancing accuracy and reducing uncertainties through its 3D capabilities. Key applications include integrating Industry Foundation Classes (IFC)-based BIM data with Machine Learning (ML) for Automated Valuation Models (AVMs), using 3D Geographic Information System (GIS) and BIM-extracted spatial attributes to refine valuation models, employing BIM for precise cost-based valuation through Detailed Replacement Cost (DRC), Quantity Takeoff (QTO), Bill of Quantities (BoQ), and Life Cycle Costing (LCC), and leveraging BIM-rendered 3D visual features for valuation through Artificial Intelligence (AI) and Computer Vision (CV). Challenges include limited access to BIM datasets, untested efficiency of 3D valuation attributes, and underutilization of BIM in cost-based valuation. Future advancements should focus on AI-driven AVMs, assessing 3D GIS and BIM impacts, and deep learning techniques for visual feature extraction, enabling greater accuracy and efficiency in property appraisal (Jafary et al., 2024). Another study highlights the integration of tree-based machine learning (ML) models enhances rent prediction accuracy compared to traditional linear regression (LR) models. Using Belgian residential rental data, random forest regression (RFR) and eXtreme gradient boosting regression (XGBR) outperformed LR, demonstrating their ability to capture nonlinear relationships. However, potential overfitting suggests caution in generalizing results. Interpretable machine learning (IML) techniques, such as SHapley Additive exPlanations (SHAP), reveal key rent determinants, including asking price, cadastral income, livable area, number of bedrooms, number of bathrooms, and proximity to points of interest. These insights help real estate professionals, policymakers, and investors make informed decisions about pricing, marketing, and property improvements. The findings confirm the significance of both structural and location factors in rent modeling while highlighting the necessity of ML models that capture nonlinearities. To enhance reliability, overfitting should be mitigated through cross-validation, regularization, pruning, or early stopping. While predictive models offer valuable insights, they should complement rather than replace expert judgment in real estate decision-making (Lenaers et al., 2024) As sustainability becomes a key priority, AI-driven models, particularly the random forest method, offer valuable insights into the impact of sustainable solutions on real estate prices. By analyzing factors such as energy efficiency, location, and environmental conditions, these models outperform traditional regression analysis by capturing nonlinear relationships. Key determinants include proximity to amenities, pollution levels, and accessibility, all of which influence property values. Despite trade-offs in interpretability, AI enhances predictive accuracy, aiding real estate professionals and investors in making informed decisions. Future advancements, including dynamic ESG data and long-term impact analysis, can further refine these models, promoting a more sustainable and data-driven real estate industry(Walacik and Chmielewska, 2024). The research focuses on forecasting property prices in China's rapidly growing housing market, which is crucial for both government policy and investment decisions. The approach employs Gaussian process regression with various kernels and basic functions to estimate the monthly pre-owned housing price index for ten major Chinese cities from March 2012 to May 2020. The models, optimized using Bayesian methods and cross-validation, predict out-of-sample price indices from June 2019 to May 2020 with relative root mean square errors ranging from 0.0458% to 0.3035% and correlation coefficients from 93.92% to 99.97%. The data was sourced from the China Real Estate Index System, which tracks various real estate categories across major cities. The developed forecasting models offer accurate predictions and can help market participants and policymakers better understand trends in the pre-owned housing sector. Future research could explore alternative Bayesian optimization strategies and expand the scope of forecasting to include more real estate price indices and cities (Jin and Xu, 2024).

2.3 Deep Learning in Real Estate Price Prediction

Deep learning has emerged as a powerful tool for predicting real estate prices, offering a sophisticated alternative to traditional methods like regression or decision trees. Real estate markets are influenced by numerous interrelated factors, such as location, property size, economic conditions, and even aesthetic elements like architecture or interior design. Deep learning models excel at capturing these complex relationships, enabling more accurate and nuanced predictions. By leveraging neural networks, deep learning allows for the integration of diverse data types, such as numerical, categorical, image, and even textual data, to generate comprehensive insights into property valuation. Despite its potential, deep learning in real estate comes with challenges. Collecting and preprocessing high-quality, diverse datasets are a significant hurdle, as real estate data often contain missing or inconsistent values. Additionally, deep learning models can be computationally intensive and difficult to interpret, creating barriers for stakeholders who require transparency in predictions. However, with advancements in explainable AI and robust data engineering practices, these challenges are being addressed, paving the way for deep learning to revolutionize the real estate sector.

In conclusion, deep learning offers transformative opportunities in real estate price prediction by integrating structured and unstructured data, capturing temporal trends, and delivering highly accurate valuations. Its ability to adapt to complex, multi-dimensional datasets positions as a game-changer for real estate analytics, empowering investors, developers, and homebuyers to make better-informed decisions in an ever-evolving market.

Artificial Intelligence (AI) and socially responsible marketing are reshaping the real estate industry by aligning transactions with social and environmental values. AI enhances innovation through data-driven insights, personalized marketing, and operational efficiency, fostering deeper connections between businesses and communities. Real estate professionals are moving beyond traditional transactions, creating emotionally compelling narratives that resonate with clients. AI strengthens customer relationships and aligns marketing strategies with stakeholder values. Integrating AI with cause-related marketing is a strategic necessity, enabling real estate businesses to support meaningful causes while maintaining profitability. This transformation positions real estate as a catalyst for positive change, where each transaction contributes to a more responsible and sustainable future (Arumugam et al., 2023). A paper introduces a novel machine learning approach that enhances standard Deep Reinforcement Learning (DRL) with time series algorithms, including Gramian Angular Field (GAF) and Long Short-Term Memory (LSTM). The model aims to provide decision support for real estate trading strategies by analyzing factors like location, loan rates, and market fluctuations, which are traditionally complex for manual analysis. The above-mentioned model was trained using historical sales data from Canberra, Australia, with attributes like "Mean," "Median," "Mode," and "Volume" analyzed across three timeframes (three months, one month, two weeks). Three DRL model variants were developed: the classic DRL, DRL with LSTM, and DRL with LSTM and GAF-optimized input data. Experimental results show that the GAF+LSTM-enhanced DRL variant outperformed others, generating more profitable investment actions but requiring more time and computational resources. The study underscores the importance of reward function design in guiding models toward better decision-making. Future research will explore integrating transformer-based architectures like BERT with DRL to improve training efficiency and parameter representation (Zhao et al., 2022) . Another study presents a unified framework for comparing deep learning models—LSTM, GRU, CNN, and their hybrid variants—for predicting the next day’s closing price of the S&P500-60 real estate index. Incorporating diverse data sources like real estate-specific indicators, macroeconomic factors, and technical metrics, the framework employs min-max normalization and reshapes data into a 3D array for sequential modeling. Models are categorized into Base (LSTM, GRU, CNN), Bidirectional/Multilayer (BiLSTM, BiGRU, Multilayer CNN), and Hybrid (CNN_BiLSTM, CNN_BiGRU, BiLSTM_BiGRU) types. Hyperparameter tuning and regularization ensure robustness, with performance evaluated using RMSE, MAPE, and correlation (R). Results identify the base GRU model with 400 neurons as the top performer (RMSE: 3.77, MAPE: 1.09, R: 0.998), followed by BiGRU (300 neurons) and BiLSTM_BiGRU in their respective categories. Validation on Vanguard Real Estate Index Fund ETF and Dow Jones U.S. Real Estate Index confirms GRU's reliability. The study highlights GRU's potential for accurate real estate index predictions and financial decision-making (Rimal et al., 2024). Similarly, another study applies Bidirectional LSTM (Bi-LSTM) networks to predict stock prices for six property and real estate indices from the LiQuid45 (LQ45) sector, using data up to May 1, 2021. Data preprocessing involved filling missing values with the last known records and splitting the dataset into an 80:20 ratio for training and testing. Feature scaling normalized the data, reshaped into a 3D array for Bi-LSTM processing. The model features a three-layer Bi-LSTM architecture, including a Bi-LSTM layer with 200 neurons, a Dropout layer (20% dropout), and a Dense layer. Compiled with the Mean Square Error (MSE) loss function and Adam optimizer, the model was trained for 20 epochs with a batch size of 32. Prediction results showed reasonable accuracy for five of the six stocks, with BSDE achieving over 90% accuracy and PTPP scoring the lowest at 8%. The average accuracy was 68%, primarily affected by PTPP's outlier performance. The Bi-LSTM model demonstrates its potential for predicting future stock values in the property and real estate sector, offering useful insights for investors and decision-makers to develop effective trading strategies (Hansun et al., 2022). Another study explores deep learning models for housing price prediction, addressing the challenge of non-linear relationships in real estate markets. Using Backpropagation Neural Network (BPNN) and Convolutional Neural Network (CNN) on a Taiwan dataset (2013–2018), it incorporates housing attributes (Type 1: "land + building," Type 2: "land + building + park") and macroeconomic indicators (e.g., housing price-to-income ratio, loan burden ratio). Findings reveal CNN achieves the highest accuracy ($R^2 > 0.945$), demonstrating its effectiveness in time-series forecasting. A five-month time window enhances predictions, while macroeconomic variables have minimal impact. Future research will investigate Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) for further advancements (Zhan et al., 2020). Another study explores using machine learning to improve real estate valuations by analyzing 1,000-bathroom images with a Convolutional Neural Network (CNN). The research shows that visual data, like interior images, can significantly impact price predictions, reducing subjectivity in traditional human evaluations. By capturing key features in images, CNNs provide a more objective, data-driven approach to pricing. Future work will focus on enhancing input processing, expanding applications to include other rooms and property types, and integrating floor plans for better accuracy. This approach aims to refine the role of ML in real estate valuation, making it more transparent and standardized (Despotovic et al., 2023). Qiao Fu (2022) explores the increasing importance of real estate tax base assessment in China’s digital economy and real estate market. The study introduces a Deep Learning Neural Network (DLNN) model to improve the efficiency and accuracy of real estate tax base assessments. The model is shown to have superior reliability, with predictions closely matching actual values, and outperforming traditional methods in terms of Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). The DLNN model offers high accuracy, fast convergence, and is well-suited for efficient batch assessments, making it capable of handling large workloads within limited time, ultimately improving work efficiency in real estate tax base evaluation (Fu, 2022) . Similarly, another study presents two frameworks for estate price prediction. The first, Basic Estate Price Prediction (BEPP), uses a Gaussian Mixture Model

(GMM) to group estate records, Space-Time Influencing Figures (STIFs) to measure the impact of surrounding facilities, and a Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) model for prediction. While accurate, the CNN-LSTM model is computationally intensive. The second framework, Lightweight Estate Price Prediction (LEPP), addresses this by ranking influential features and using a lightweight shallow Recurrent Neural Network (RNN) for faster predictions, making it suitable for online queries. Both frameworks were validated with Taiwanese real estate data, demonstrating efficient and accurate price prediction (Chiu et al., 2022). Another study focuses on predicting which of two exterior images of the same property is more attractive, as people's perceptions can vary greatly based on the image capture method and angle. Traditional models often regress an exact attractiveness score, but evaluating attractiveness stably using exact values is challenging. Instead, the proposed model predicts which of two images is more attractive without requiring an exact attractiveness value. A dataset of real-estate images from At Home Co., Ltd. was created, and attractive annotations were made using a pairwise comparison experiment. The model achieved higher accuracy than conventional methods by considering both similar and different attractiveness levels through learning to rank and score regression. The accuracy improved as the attractiveness difference between images increased, reaching at least 85% accuracy for pairs with the greatest attractiveness difference. This pairwise comparison learning network can also be applied to general image attractiveness ranking tasks. Future work will focus on visualizing and improving the factors that contribute to perceived image attractiveness (Wang et al., 2019). Another study applies Deep Learning (DL) to real estate mass appraisal, evaluating various Neural Network (NN) architectures and tuning parameters. Using Swiss residential apartment transaction data from Fahrländer Partner Raumentwicklung (FPRE) (88,247 samples from 2011–2017), it compares DL with Linear Regression (LR), Gradient Boosted Trees (GBT), and a Meta Model (MM) combining all approaches. Results show DL outperforms LR but slightly lags behind GBT in predictive accuracy. However, MM achieves the best performance with a Mean Squared Error (MSE) of 0.0332, outperforming DL (0.0373), GBT (0.0367), and LR (0.0383). While DL shows promise, some overfitting is observed. The study explores different NN architectures, considering layer composition, spatial effects, hidden layers, and regularization. Future work could address sample imbalance across districts and enhance spatial and temporal modeling in DL frameworks (Walthert et al., n.d.). Similarly, another study highlights the significance of real estate price prediction in economic development, aiding investment strategies, risk management, and urban planning. This systematic review classifies prediction models into Machine Learning (ML), Deep Learning

(DL), and Hybrid Models (HM) based on an extensive search across Google Scholar and Scopus. The findings indicate that ML (61.5%) is widely used due to its adaptability but faces challenges with unstructured data, while DL (7.7%) effectively handles complex data but requires high computational resources. HM (one-third of studies) offers the best predictive accuracy by integrating ML and DL. Despite the rising interest in DL and HM, research gaps remain, particularly in computational efficiency and handling unstructured data. Future studies should focus on optimizing algorithms and leveraging diverse data sources to enhance real estate price prediction accuracy, benefiting both researchers and industry professionals (Naz et al., 2024) . Similarly, another study explores the application of Deep Neural Networks (DNNs) in real estate price prediction, particularly for small datasets where high dimensionality reduces accuracy. To address this, Principal Component Analysis (PCA) is integrated with DNN (PCA-DNN) to enhance prediction by reducing dimensionality, transforming datasets, and localizing key price-influencing features. Using real estate data from Sahibinden.com (2021) for 42 districts in Trabzon, Türkiye, the PCA-DNN model achieves 90%–95% accuracy, outperforming standard DNN and Stepwise Regression Analysis DNN (SRA-DNN) models. Findings highlight that spatial attributes and building age are the most influential factors in price determination. The study underscores PCA's role in improving DNN performance, especially where data availability is limited. Future research should explore variations like Kernel PCA and Robust PCA to account for time-varying price trends and inflation factors (Mostofi et al., 2022). Another study investigates house price prediction using Deep Learning (DL) techniques, focusing on Convolutional Neural Networks (CNNs) with hyperparameter tuning for improved accuracy. The study highlights the importance of sufficient data in predicting real estate prices accurately, benefiting property owners, contractors, real estate firms, and financial institutions. The proposed model optimizes Learning Rate (LR) by testing seven optimizers on a house sales price dataset, evaluating their impact across multiple epochs. The best-performing optimizer, Root Mean Squared Logarithmic Error (RMSLE)-based Root Mean Square Propagation (RMSprop), achieves the lowest Error Rate (ERR) (14.048) and the highest accuracy in R-squared (R^2) (0.8852) and Adjusted R^2 (0.8841). Findings confirm that optimizer sensitivity significantly enhances CNN-based price prediction models. Future research should explore hyperparameter tuning in Machine Learning (ML) models using RandomizedSearchCV to further improve accuracy,(K. Radhakrishnan et al / Analyzing House Sales Prices byhyperparameters tuning Method Using Deep Learning (DL) Techniques Analyzing House Sales Prices byhyperparameters tuning Meth, n.d.) . A study on real estate appraisal explores the impact of property characteristics, location, and neighborhood on pricing. Traditional Computer-Assisted Mass Appraisal (CAMA) models use hedonic pricing models and structured data but incorporating Geographic Information Systems (GIS) and Convolutional Neural Networks (CNNs) can enhance accuracy. Analyzing 71,000 properties from OpenDataPhilly (2011), the study includes housing size, quality, technical details, amenities, and ZIP code effects. Results show geospatial data (proximity to parks, schools, traffic noise) improves accuracy by 12%, while image data (house style, aesthetics) increases it by 16%. Combining both yields only 8% improvement in Mean Absolute Error (MAE) due to overlapping signals but reduces extreme errors in Root Mean Square Error (RMSE). A complex multi-input neural network struggled due to optimization challenges. The study concludes that geospatial and image data enhance property valuation, offering valuable insights for real estate pricing models (Kucklick et al., 2021). Enterprise credit risk prediction assesses the likelihood of future defaults by analyzing historical operational data. To enhance prediction accuracy for listed real estate enterprises and support government management, an attention-based Convolutional Neural Network-Bidirectional Long Short-Term Memory (CNN-BiLSTM) hybrid model, integrated with logistic regression features, was developed. Using financial data from 2017 to 2020 for real estate enterprises listed on China's Shanghai and Shenzhen stock exchanges, the model achieved an average sensitivity of 99.28%, specificity of 94.57%, and quality index of 97.15%, outperforming previous models such as Particle Swarm Optimization-Support Vector Machine (PSO-SVM), Rough Set-PSO-Support Vector Regression (RS-PSO-SVR), and PSO-Backpropagation (PSO-BP). Key financial indicators, including profitability, debt-paying ability, growth ability, operational efficiency, and basic enterprise information, were refined through variance inflation factor analysis and logistic regression to construct a scientific credit risk prediction index system. By integrating a one-dimensional CNN, BiLSTM network, and attention mechanism, the model demonstrated superior predictive performance with an F1-score of 96.85%. It serves as a valuable tool for market participants and government regulators, particularly during periods of real estate market volatility. Future research may incorporate non-financial indicators, macroeconomic variables, and interest rate trends to enhance the model's generalization ability across different industries and small-sample datasets (Zhang et al., 2024). Accurate real estate assessments are essential for determining property taxes in smart cities and avoiding disputes between property owners and local governments. A deep learning approach using a structured Deep Neural Network (DNN) based on a layered knowledge graph is proposed to improve property valuation. The model is designed to be time- and space-efficient, requiring fewer data points for training and adapting to market trends. A case study using Zillow data showed that this structured DNN outperforms traditional methods like multivariate linear regression and fully connected neural networks, providing more accurate house price predictions. The approach addresses inefficiencies in deep learning by limiting unnecessary connections, reducing training time, and minimizing overfitting. Future research will focus on automating knowledge graph extraction, developing scalable DNNs that adapt to market changes, and extending the method to mobile cloud computing for real estate assessments, with potential applications in other sectors like stock markets, healthcare, and transportation (2017 IEEE SmartWorld : Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI) : 2017 conference proceedings : San Francisco Bay Area, California, USA, August 4-8, 2017, 2017). The real estate auction market plays a crucial role in financial and investment decisions, yet few artificial intelligence-based studies have focused on predicting auction prices. Forecasting models using regression, artificial neural networks, and genetic algorithms (GA) were developed to predict real estate auction prices, specifically for apartments in Seoul between 2013 and 2017. The GA model, enhanced by regional segmentation based on auction appraisal prices, outperformed other models in terms of accuracy. The findings emphasize the importance of the grouping process for improving prediction performance, offering valuable insights for investors and real estate fund managers. The results contribute to the efficiency of real estate auction markets and economic growth. However, the model is limited to Seoul's apartment auction market during the specified time frame, and future work could expand its application to other real estate sectors and incorporate more diverse data for better forecasting (Kang et al., 2020). China’s rapidly growing housing market demands accurate forecasting to support investors and policymakers. Using data from the China Real Estate Index System (CREIS), a Neural Network (NN) model was applied to predict office property price indices in 10 major cities from July 2005 to April 2021. A simple NN with three delays and three hidden neurons achieved stable results, with an average Relative Root Mean Square Error (RRMSE) of 1.45% across training, validation, and testing. The model can function alone or alongside traditional forecasting methods. Future directions include combining econometric time series models with Machine Learning (ML), such as integrating cointegration analysis with NNs and transitioning from univariate to multivariate models to boost accuracy and applicability (Xu and Zhang, 2024) .

2.4 Summary

Machine learning and deep learning are transforming real estate price prediction by offering superior accuracy, efficiency, and the ability to model complex, non-linear relationships across diverse data types. In dynamic markets like Dubai, these technologies help forecast property values, identify investment opportunities, and support urban planning initiatives, including smart housing and energy-efficient infrastructure. Deep learning models—such as CNNs, LSTMs, and GRUs—excel in analyzing both structured and unstructured data, including images and text, to enhance valuation accuracy and guide data-driven investment strategies. While challenges such as data transparency, computational demands, and model interpretability persist, advancements in explainable AI and lightweight architectures are addressing these issues. Overall, AI-driven approaches are enabling more sustainable, ethical, and informed decision-making in the real estate industry.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

This study focuses on Dubai real estate price prediction using advanced Machine learning and Deep learning models. Predicting prices is useful for investors, buyers, sellers, real estate brokers to help them make data driven decisions. Real estate prices are shaped by various factors, including location, property size, market trends, economic conditions, and more. Advanced ML models are effective at managing large, multidimensional datasets and identifying complex interactions among these variables. These models are designed to learn from historical data, enabling them to make highly accurate predictions and adapt to fluctuations in market conditions. Moreover, advanced ML models offer enhanced generalization and robustness. Advanced deep learning (DL) models excel at capturing complex, non-linear relationships between a wide range of factors influencing real estate prices, such as location, property size, market trends, and economic conditions. Unlike traditional models, DL models can process both structured and unstructured data, including numerical property attributes, images, and textual descriptions, allowing for a more holistic analysis. They are particularly effective for analyzing time-series data, making them ideal for predicting price trends over time. Additionally, DL models scale efficiently with large datasets, deliver high prediction accuracy, and adapt swiftly to evolving market conditions, making them highly valuable for real estate forecasting.

The key point of this study is that a comparative analysis between advanced ML and DL models. The ML models include Support Vector Regression (SVR), Random Forest, Gradient Boosting Regression (GBM), Extreme Gradient Boosting (XGBoost), Ridge and Lasso. Each model has its own unique feature. SVR is effective in high dimensions, robust to overfitting, and adaptable through kernel functions. Random Forest has advantages like robustness and efficiency in feature selection and handling missing values. Also, GBR is popular for high predictive accuracy, versatile in application, capable of capturing feature interactions, and incorporates regularization techniques. XGBoost also has its own features like handling missing values and using tree pruning. Similarly, Ridge and Lasso can handle multicollinearity and regularization features.

The DL models include Transformer, TabTransformer, Feature-Token Transformer (FT-Transformer) and Gated Recurrent Units (GRU), Deep and Cross Networks (DCN).

TabTransformer is used for tabular data, effectively learns complex feature representations, adapts seamlessly to various tasks, and manages categorical features proficiently. Similarly, FT-Transformer improves generalization and scalable. Finally, GRU is used for faster training and addressing the vanishing gradient problems.

Each model comes with its own set of advantages, making them suitable for different aspects of real estate price prediction. To determine the most effective model, it is essential to implement and evaluate their performance using key evaluation metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R-squared. MAE provides a straightforward measure of the average magnitude of errors in a set of predictions, giving insight into how closely the predicted values align with actual prices. RMSE, on the other hand, emphasizes larger errors by squaring the differences, making it particularly useful for identifying significant discrepancies in predictions. R-squared offers a statistical measure of how well the model explains the variability of the dependent variable, indicating the proportion of variance in real estate prices that is predictable from the independent variables. By systematically analyzing the results from these evaluation metrics, we can identify which model performs best, enabling stakeholders to make informed decisions based on the most accurate price predictions.

3.2 Structure of Proposed Framework

The proposed framework for predicting Dubai real estate prices utilizes advanced machine learning (ML) and deep learning (DL) models to effectively analyze the structured dataset provided by the Dubai Land Department (DLD). This dataset consists of categorical features (such as property location, transaction type, and property category) and numerical attributes (including property size, transaction date, and sale price). Given the structured nature of the data, capturing both linear and non-linear relationships is crucial for accurate price forecasting. The framework employs ML models like Support Vector Regression (SVR) and Random Forest, which efficiently model structured data and identify feature importance. However, to enhance predictive accuracy, it integrates DL models such as TabTransformer and Gated Recurrent Units (GRUs). TabTransformer is specifically designed for structured data, leveraging attention mechanisms to optimize feature interactions, while GRUs excel in capturing sequential trends, making them valuable for time-based price fluctuations. This hybrid approach combines the strengths of both ML and DL techniques, ensuring robust price predictions across diverse property types and market conditions. By leveraging advanced modeling techniques and structured data processing, this framework provides data-driven insights to support investors, developers, and policymakers in Dubai’s dynamic real estate market.

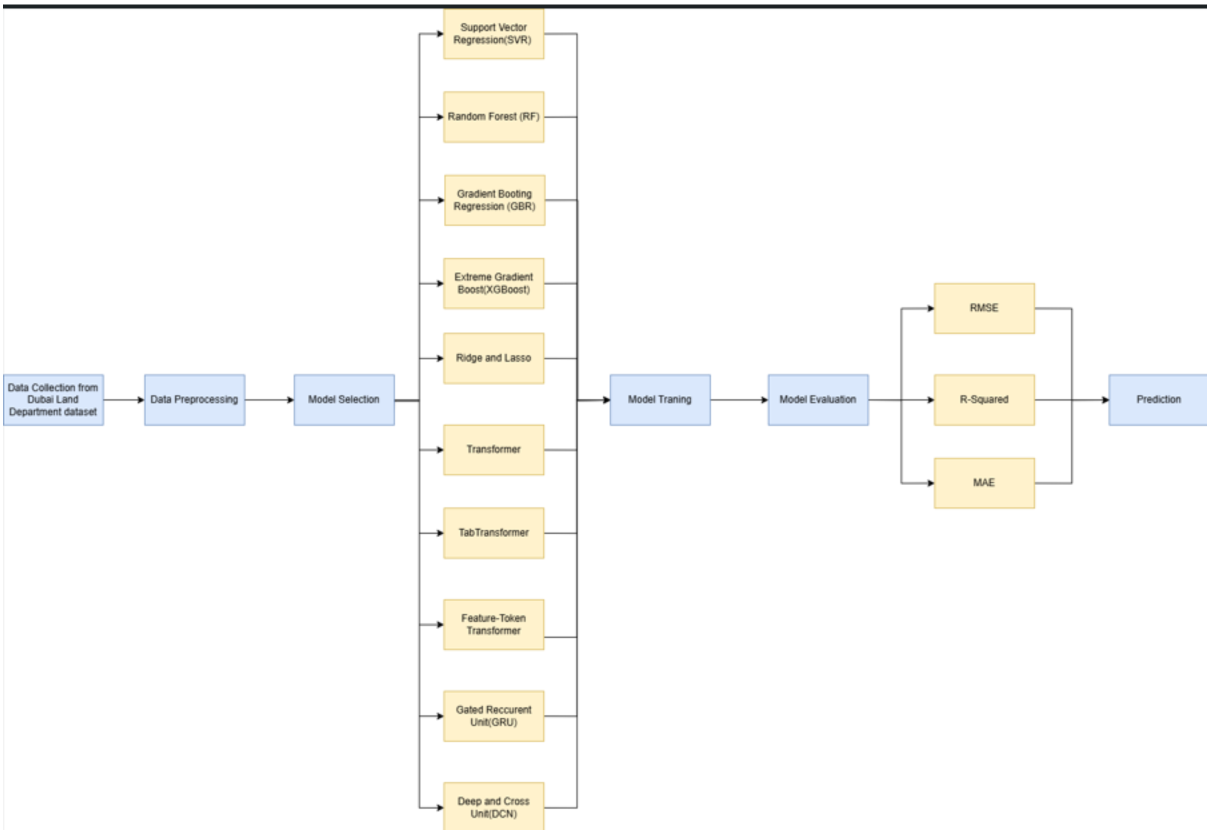

Data Collection from Dubai Land Department dataset
Data Preprocessing
Model Selection
Support Vector Regression(SVR)
Random Forest (RF)
Gradient Booting Regression (GBR)
Extreme Gradient Boost(XGBoost)
Ridge and Lasso
Transformer
TabTransformer
Feature-Token Transformer
Gated Reccurent Unit(GRU)
Deep and Cross Unit(DCN)
Model Traning
Model Evaluation
RMSE
R-Squared
MAE
Prediction

Fig 1. Proposed Research Framework

The prediction process begins with collecting real estate transaction data from the Dubai Land Department (DLD) and preprocessing it to handle outliers and prepare features for modeling. The dataset is then split into training and testing sets, with property price as the target variable. Machine learning (ML) models such as SVR, Random Forest, GBR, XGBoost, Ridge, and Lasso Regression undergo hyperparameter tuning for optimization, while deep learning (DL) models like Transformer, TabTransformer, FT-Transformer, GRU, and DCN leverage advanced architectures to capture complex relationships within the data. Model performance is assessed using key evaluation metrics, ensuring accurate and reliable real estate price predictions suited to Dubai’s dynamic market. Then model performance is evaluated using key metrics such as R², RMSE, and MAE to assess accuracy and reliability in predicting property prices as shown in the above figure. Each step is explained below.

3.2.1 Dataset Collection

The dataset from the Dubai Land Department, available on Dubai Pulse (link: https://www.dubaipulse.gov.ae/data/dld-transactions/dld_transactions-open), provides comprehensive data on real estate transactions in Dubai. It includes various features that are crucial for predicting property prices and analysing market trends. The key attributes in the dataset are:

1. Transaction Date: The date of the real estate transaction, crucial for understanding market trends over time.
2. Property Location: Geographical information about where the property is located, a significant factor influencing price.
3. Property Size: The size of the property, typically measured in square feet or square meters, which directly affects the transaction price.
4. Transaction Type: Indicates whether the transaction was a sale, mortgage, or other types, providing insights into the nature of real estate activities.
5. Property Type: Describes the type of property (e.g., apartment, villa, land), which plays a crucial role in price determination.
6. Number of Bedrooms: For residential properties, the number of bedrooms is a key feature impacting the value.
7. Transaction Price: The target variable, representing the price at which the property was transacted.

This dataset contains a mix of categorical and numerical variables, making it ideal for applying both traditional Machine Learning and Deep Learning models. The categorical features, such as location, property type, can be embedded, while the numerical features, such as property size, transaction price, are normalized for modeling purposes. It provides rich, structured data for analyzing real estate trends and predicting property prices in Dubai's dynamic market.

3.2.2 Data Preprocessing

Data preprocessing is an essential step in ensuring the dataset from the Dubai Land Department is clean, structured, and ready for use in machine learning (ML) and deep learning (DL) models. The preprocessing steps include:

3.2.2.1 Handling Missing Data

Missing values are a common issue in real estate datasets. For this dataset, missing values in numerical features (such as property size or transaction price) are either imputed using statistical methods like mean or median, or they are removed if the missing data proportion is too large. For categorical features (e.g., property type, transaction type), missing values can be filled using the mode or treated as a separate category.

3.2.2.2 Categorical Feature Encoding

Categorical variables such as property location, transaction type, and property type need to be converted into numerical form to be processed by ML and DL models. This is done using techniques like **one-hot encoding** for ML models or **embedding layers** for deep learning models, which allow the model to capture relationships between categories.

3.2.2.3 Normalization/Standardization

The numerical features (e.g., property size, transaction price, transaction date) are normalized or standardized to ensure that all features are on a similar scale. Normalization (scaling values between 0 and 1) or standardization (scaling based on mean and standard deviation) helps improve model convergence during training and ensures that no single feature disproportionately affects the model's predictions.

3.2.2.4 Outlier Detection and Treatment

In real estate data, outliers (e.g., extremely high or low property prices or unusually large property sizes) can distort model predictions. Outliers are detected using statistical methods such as Z-scores or the interquartile range (IQR) and are either removed or capped to reduce their impact on model performance.

3.2.2.5 Date Feature Engineering

The transaction date can be transformed into more meaningful features such as the year, month, or even quarter to capture seasonal or market trends. For instance, the real estate market may show patterns based on the time of year, and this information can be used to improve the predictive power of the model.

3.2.2.6 Feature Selection

Not all features may be relevant for predicting property prices. Feature selection techniques such as correlation analysis or algorithms like Lasso regression can help identify the most important features. This step reduces dimensionality, simplifies the model, and improves performance.

3.2.3 Model Selection

The model selection process involves evaluating various ML and DL models to identify the most effective for real estate price prediction. ML models like SVR, Random Forest, GBR, XGBoost, Ridge, and Lasso capture linear, non-linear, and ensemble-based patterns while managing overfitting. DL models such as Transformers, TabTransformer, Feature-Token Transformer, GRU, and DCN leverage attention mechanisms, tokenization, and sequential processing to capture complex dependencies. By comparing these models, the best-performing one is selected based on predictive accuracy and generalization ability.

3.2.4 Model Training

Once the dataset is preprocessed and the models are selected, the training phase begins. Each chosen model is trained using historical real estate transaction data, allowing it to learn patterns and relationships between property features and prices. Machine learning models, such as SVR, Random Forest, GBR, XGBoost, Ridge, and Lasso, adjust their parameters to capture trends in the data, while deep learning models, including Transformers, TabTransformer, Feature-Token Transformer, GRU, and DCN, leverage neural networks to extract complex interactions and dependencies. During training, the models optimize their parameters to minimize errors and enhance predictive performance, ensuring they generalize well to unseen data.

3.2.5 Model Evaluation

The trained models are assessed using key performance metrics:

- Root Mean Squared Error (RMSE): Measures the average prediction error.
- R-Squared (R^2): Indicates the proportion of variance explained by the model.
- Mean Absolute Error (MAE): Evaluates the average absolute prediction error.

3.2.6 Prediction

The model that demonstrates the highest accuracy and reliability during evaluation is selected for real estate price prediction. Using newly provided property features, this model generates price estimates by applying the learned relationships from historical data. By using its optimized parameters and pattern recognition capabilities, the selected model ensures precise and data-driven predictions, aiding stakeholders in making informed real estate decisions.

3.3 Research Overview

The research methodology in this study employs various advanced Machine Learning (ML) and Deep Learning (DL) models to predict Dubai real estate prices, aiming to identify which model outperforms the others. The ML models utilized include Support Vector Regression (SVR), Random Forest, Gradient Boosting Regression (GBR), Extreme Gradient Boosting (XGBoost), Ridge, and Lasso regression. For DL models, the study implements Transformer, TabTransformer, Feature-Token Transformer (FT-Transformer), and Gated Recurrent Unit (GRU), Deep and Cross Networks (DCN). Model performance is evaluated using the metrics Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-Squared (R^2).

3.3.1 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a powerful supervised machine learning model that handles regression tasks by finding a function that best approximates the relationship between input features and target values while maintaining a margin of tolerance. In this study, we applied SVR to predict Dubai real estate prices, leveraging its ability to handle both linear and non-linear relationships effectively. In a particular study, EEMD-SD-SVM and SVM demonstrated competitive performance, especially in cases where establishing clear decision boundaries was essential. These models offer a robust solution for property price prediction when precise separation between data points is critical (Elnaeem Balila and Shabri, 2024). The dataset D= $\{(x_1, y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$ represents the features of properties $x_i \in R^d$ (such as property size, location, and type) and the real estate price $y_i \in R$ (the target variable that is price of the property)

The objective of SVR is to determine a function **f(x)** that predicts the target value **y** within a specified tolerance ϵ, while maintaining model simplicity.

SVR defines the prediction function as:

f(x)=w·x+b (1)

Where,

- f(x) is the predicted price,
- w is the weight vector (which defines the hyperplane),
- x is the input feature vector (e.g., size, location),
- b is the bias (intercept term).

SVR solves the following optimization problem to find w and b:

$$min\frac{1}{2}\|w\|^2C\sum_{i=0}^{n}(\xi_i+\xi_i^*) \quad (2)$$

Subject to,

$$|y_i - w * x_i - b| \leq \in +\xi_i \quad (3)$$

Where:

- $\xi_{i,}\xi_i^*$ are slack variables that allow some data points to fall outside the ϵ-tube (i.e., the error margin).
- C is the regularization parameter that controls the trade-off between maximizing the margin and minimizing the prediction error.
- ϵ defines the margin of tolerance, within which predictions are considered acceptable.
- ||w||^2 is the regularization term that helps control overfitting by penalizing large weights.

The interpretation of the equation is, $\frac{1}{2}$||w||^2 minimizes the norm of the weight vector, ensuring a flatter, lower-complexity model. The slack variables $\xi_{i,}\xi_i^*$ account for data points that fall outside the ϵ-margin, allowing for some error. The parameter C controls the trade-off between fitting the data closely and margin width. A large C forces the model to fit the data more precisely, which may lead to overfitting, while a smaller C allows for a wider margin and more tolerance for errors.

Once w and b are found, the predicted price for a new property,

$\hat{y}$= w·x$_{new}$+b (4)

Finally, the model performance can be calculated by MAE, RMSE and R^2.

3.3.2 Random Forest (RF)

Random Forest is a robust machine learning algorithm that creates multiple decision trees during training, each using a random subset of the dataset and features. This randomness reduces overfitting and enhances prediction accuracy. For predictions, it aggregates the results of all trees, either by voting (for classification) or averaging (for regression). Known for handling complex data and providing reliable forecasts, Random Forest is widely used for both classification and regression tasks. In this study RF is used for regression tasks. The Random Forest algorithm showcased strong predictive power and adaptability, effectively managing complex, noisy data while consistently delivering accurate property price predictions (Elnaeem Balila and Shabri, 2024).

The RF uses the features in the Dubai real estate transaction dataset to build the decision trees. RF randomly select the features like property type, location, property size to split the nodes. This ensures that each tree gets a slightly different view of the data, leading to more robust predictions. This helps prevent overfitting and allows RF to capture a variety of patterns in the data. For each subset of data, a decision tree is grown. Each tree makes a prediction on the target variable, in this study, the real estate price, by splitting the data based on the selected features. Once all trees have been grown, the final prediction is made by averaging the predictions from all the individual decision trees. The mathematical representation as follows,

$$y = \frac{1}{T}\sum_{t=1}^{T} y_t \tag{5}$$

Where,

- y is the final predicted price.
- T is the number of decision trees.
- y_t is the predicted price from the t-th tree.

3.3.3 Gradient Boosting Regression (GBR)

Gradient Boosting Regression (GBR) is a powerful ensemble learning technique designed for regression tasks, such as predicting real estate prices. It builds models sequentially, where each new model corrects the residual errors of the previous one using gradient descent optimization. GBR employs decision trees as weak learners, refining predictions iteratively to improve accuracy. A study on house price prediction during COVID-19 compared various models including GBM, Random Forest, and Extra Tree Regressor. The results showed that GBM exhibited the least overfitting and provided the fastest price predictions compared to the other models (Mora-Garcia et al., 2022).

For a regression task like Dubai real estate price prediction, GBR starts by initializing the model with the mean of the target variable:

$$f_0(x) = mean(y) \tag{6}$$

Where y is the target variable i.e. property price.

GBR operates iteratively, where each iteration involves training a new model (typically a decision tree) to predict the residual errors of the previous model. The goal is to minimize the loss function $L(y, \hat{y})$, where y is the actual price and $\hat{y}$ is the predicted price. In each iteration m, the model computes the residuals (errors) as the difference between the true value y and the predicted value $\hat{y}$ from the previous iteration.

$$r_i^m = y_i - f_{m-1}(x_i) \tag{7}$$

Where,

- r_i^m is the the residual for each sample.
- y_i is the actual value (property price).
- $f_{m-1}(x_i)$ is the predicted value from the previous iteration.

A new decision tree (weak learner) $h_m(x)$ is trained to predict the residuals from the previous iteration. This model aims to correct the errors made by the prior models. The updated model is then added to the previous model's prediction with a learning rate α, which controls the contribution of the new model:

$$f_m(x) = f_{m-1}(x) + \alpha h_m(x) \tag{8}$$

Where α is a hyperparameter that adjusts the step size in the gradient descent process. A small α usually prefer to avoid overfitting.

The process continues for a set number of iterations or until the error is minimized to an acceptable level. Each new model works to correct the errors made by the ensemble of previous models. Once all iterations are complete, the final prediction is the combined sum of the predictions from all the individual models.

$$\hat{y} = f_M(x) = f_0(x) + \sum_{m=1}^{M} \alpha h_m(x) \tag{9}$$

Where,

- M is the total number of iterations.

The loss function of the GBM model that typically used in regression tasks is squared loss function, that is,

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2 \tag{10}$$

3.3.4 Extreme Gradient Boosting (XGBoost)

XGBoost is an advanced machine learning algorithm within the ensemble learning category, based on the gradient boosting framework. It uses decision trees as base learners and incorporates regularization techniques to improve model generalization. Known for its computational efficiency, XGBoost excels at processing large datasets quickly, provides insightful feature importance analysis, and effectively handles missing values. It's a popular choice for various tasks, including regression, classification, and ranking. On a study of housing price prediction, XGBoost significantly outperformed NNGP (Nearest Neighbor Gaussian Processes), especially with larger sample sizes (n = 106). It achieved the highest prediction accuracy across all sample sizes, error measures, price bands, and on both logarithmic and real scales. XGBoost uses sequential learning, where each tree builds on the previous one's results, leading to improved predictions. Its superior accuracy over NNGP may be attributed to the skewed distribution of the dependent variable (Mora-Garcia et al., 2022).

The process begins with an initial prediction, usually a constant value such as the mean of the target variable that is price of the property. Initial model is,

$$f_0(x) = mean(y) \tag{11}$$

In each subsequent iteration m, the model calculates the residuals (errors) by taking the difference between the actual property prices and predicted from the previous model.

$$r_i^m = y_i - f_{m-1}(x_i) \tag{12}$$

Where,

- r_i^m is the residual of the ith property of the mth iteration.
- $f_{m-1}(x_i)$ is the prediction from the previous iteration.

In each iteration, a new decision tree (base learner), denoted as $h_m(x)$, is trained to model the residuals, which represent the errors from the previous iteration's predictions. The goal of this tree is to capture and correct the differences between the actual property prices and the predictions made by the prior model. By focusing on minimizing these residuals, the decision tree refines the predictions, effectively reducing the overall error in the model. As more trees are added, each one learns from and corrects the mistakes of the previous trees, leading to progressively better and more accurate property price predictions.

$$h_m(x) = argmin_h \sum_{i=1}^{n}(r_i^{(m)} - h(x_i))^2 \tag{13}$$

The predictions generated by the new decision tree are incorporated into the overall model to improve its accuracy. However, to prevent overfitting, where the model becomes too specialized to the training data and loses generalization capability, a learning rate (denoted as α) is applied. The learning rate is a crucial hyperparameter that controls the degree to which the new decision tree contributes to the model's overall prediction. By scaling down the impact of each new tree, the learning rate ensures that the model improves in a gradual and controlled manner, rather than over-adjusting to the data. This approach balances the trade-off between model complexity and accuracy, preventing overfitting while maintaining robust performance on unseen data. The update to the model can be represented mathematically as:

$$f_m(x) = f_{m-1}(x) + \alpha h_m(x) \tag{14}$$

XGBoost optimizes an objective function that strikes a balance between maximizing the model's performance and minimizing its complexity. This objective function is composed of two key components: the loss function L and the regularization term *Ω*. The loss function measures the prediction error, which quantifies how well the model's predictions match the actual values. In the context of real estate price prediction, this error represents the difference between the predicted property prices and the actual prices in the dataset. XGBoost commonly uses the mean squared error (MSE) for regression tasks, but other loss functions may also be used depending on the problem at hand. The goal is to minimize this error over all iterations, improving the model's accuracy in predicting property prices. While minimizing the loss function improves the model's fit to the data, there's a risk of overfitting, where the model becomes too complex and performs well on the training data but poorly on new, unseen data. To prevent overfitting, XGBoost incorporates a regularization term *Ω*, which penalizes overly complex models. This regularization term controls the complexity of the decision trees used by XGBoost, such as by limiting the depth of the trees or the number of leaves. By including this penalty, XGBoost encourages simpler models that generalize better to new data. Objective function as follows,

$$Objective = \sum_{i=1}^{n} L(y_i, \hat{y_i}) + \sum_{k=1}^{K} \Omega(h_k) \tag{15}$$

Where,

- $\Omega(h_k)$ is the regularization term which control the complexity of the tree and avoid overfitting.

After a pre-determined number of iterations or when the model converges, the final prediction for the property prices $\hat{y}$ is the sum of all the base learners (trees):

$$\hat{y} = f_M(x) = f_0(x) + \sum_{m=1}^{M} \alpha h_m(x) \tag{16}$$

Where,

- M is the total number of iteration the model used

3.3.5 Ridge and Lasso

Ridge and Lasso regression are both types of regularization techniques used to improve linear regression models by preventing overfitting, especially when dealing with complex datasets like the Dubai real estate transaction dataset. Both approaches add a penalty term to the loss function, but they differ in how they apply this penalty.

3.3.5.1 Ridge Regression

Ridge regression applies an L2 penalty, which is the squared magnitude of the model coefficients, to the loss function. This means that while the model minimizes the sum of squared residuals (as is done in ordinary least squares regression), it also incorporates an additional penalty term that is proportional to the sum of the squared values of the model coefficients. The inclusion of this penalty helps to prevent the coefficients from becoming too large, which can occur when the model is overfitting the training data. In the study on predicting continuous data using the Boston House Pricing dataset, Ridge Regression with cross-validation achieved an impressive R^2 of 88.3%. This regularization method effectively controlled model complexity, reducing overfitting. Cross-validation helped optimize the regularization parameters, enhancing both the model's predictive performance and its ability to generalize to unseen data (Aljuboori and Abdulrazzq, n.d.).

The goal of Ridge regression is not only to achieve a good fit for the training data but also to ensure that the model generalizes well to new, unseen data. This is particularly important in situations where the dataset contains features that are highly correlated with one another, a condition known as multicollinearity. In such cases, traditional linear regression may produce large, unstable coefficient estimates, leading to poor performance on test data. Ridge regression mitigates this issue by shrinking the coefficients, thereby improving model stability and preventing overfitting.

The Ridge Regression loss function is,

$$f = min_{\beta}(\sum_{i=1}^{n}(y_i - \hat{y})^2 + \lambda \sum_{j=1}^{p} \beta_j^2) \quad (17)$$

Where,

- y_i is the actual price of the property.
- $\hat{y}$ is the predicted price from the model.
- β_j are the coefficients (weights) for each feature j in the model (e.g., property size, location, number of bedrooms).
- λ is a regularization parameter that controls the strength of the penalty.
- n is the number of observations (properties) in the dataset.
- p is the number of features.

In the Dubai real estate dataset, key features such as property size, location, number of bedrooms, transaction date, and other property-specific details are used to predict the property price. These features capture the various factors influencing real estate prices in Dubai, a market driven by location demand, property specifications, and market timing. When applying Ridge regression to this dataset, the goal is to model the relationship between these features and property prices while preventing overfitting. Overfitting happens when a model learns irrelevant patterns or noise in the training data, resulting in poor performance on new data. Ridge regression tackles this by introducing an L2 regularization penalty to the loss function, shrinking the coefficients of less important features. This prevents any one feature from disproportionately influencing the model, thus ensuring a more balanced and stable prediction. For instance, in a dataset where features like property size and location are highly correlated, traditional linear regression might assign overly large coefficients to these variables, potentially harming the model's accuracy on unseen data. Ridge regression reduces this risk by shrinking these coefficients, ensuring that no single variable dominates the model. As a result, the model becomes more stable and adaptable, producing reliable predictions across various transactions.

Ridge regression also helps generalize the model to new property transactions, an essential feature in Dubai’s dynamic real estate market where prices fluctuate frequently due to multiple factors. Regularization features buffer the model against noise and variability, such as outliers or incomplete data, allowing it to deliver consistent predictions across different types of properties, locations, and market conditions. As the real estate market grows and the dataset expands, Ridge regression’s ability to handle many features while maintaining model simplicity makes it highly scalable. It remains a strong choice for accurately predicting property prices in Dubai, offering flexibility, stability, and adaptability in a fast-evolving market

3.3.5.2 Lasso Regression

Lasso regression uses an L1 penalty (absolute value of coefficients) instead of L2. Unlike Ridge, which shrinks coefficients without making them zero, Lasso can shrink some coefficients to exactly zero, effectively performing feature selection by removing irrelevant or redundant features from the model. This makes Lasso particularly useful when dealing with datasets with many features, as it simplifies the model. The results of the LASSO regression model are relatively effective, as demonstrated by the data. However, the study was limited by a small sample size and only involved internal validation, which impacted the model's generalizability to some extent. In future research, we aim to incorporate larger sample sizes and external validation on different populations to further enhance the model's predictive performance and generalization ability (Wang et al., n.d.).

The Lasso Regression loss function is,

$$f = min_{\beta}(\sum_{i=1}^{n}(y_i - \hat{y})^2 + \lambda \sum_{j=1}^{p}\left|\beta_j\right|) \tag{18}$$

Where,

- y_i is the actual price of the property.
- $\hat{y}$ is the predicted price from the model.
- $\left|\beta_j\right|$ is an L1 penalty on the coefficients.
- λ is a regularization parameter that controls the strength of the penalty.

- n is the number of observations (properties) in the dataset.
- p is the number of features.

In the context of the Dubai real estate transaction dataset, Lasso regression is highly effective in selecting the most critical features for predicting property prices. By applying an L1 regularization penalty to the loss function, Lasso has the unique ability to shrink certain feature coefficients to exactly zero, effectively excluding irrelevant or less impactful predictors. This process results in a simpler, more interpretable model by concentrating only on the most important variables, such as location, property type, and transaction type. For example, in a dataset with various attributes like property size, year of construction, number of bedrooms, floor level, and market trends, not every feature will have a substantial effect on property prices. Lasso regression automatically assigns zero coefficients to features that do not significantly enhance the model's predictive performance, thereby creating a more focused model that prioritizes variables, such as prime locations or specific transaction types, which are key drivers of property values. This feature selection is particularly valuable in real estate, where datasets often include a multitude of attributes, some of which may introduce noise rather than improve the model. By narrowing the focus to key features, Lasso reduces the complexity of the model and minimizes overfitting, allowing it to perform better on unseen data. A more streamlined model also enhances interpretability, making it easier for stakeholders like investors, brokers, and policymakers to understand the factors that most influence real estate prices in Dubai.

In a fast-changing market like Dubai, where prices are affected by factors ranging from infrastructure developments to global economic conditions, Lasso's ability to filter out irrelevant features becomes even more important. By homing in on the most impactful variables, Lasso regression helps deliver more accurate and reliable predictions across diverse property types and market conditions, empowering stakeholders to make more informed and confident decisions.

3.3.6 Transformer

The Transformer model effectively processes the DLD real estate dataset by leveraging Multi-Head Attention to capture interactions between numerical and categorical features. The dataset is first transformed into a unified input representation X, where numerical features are normalized, and categorical features are embedded into continuous vector spaces. This enables the model to learn meaningful relationships across diverse property attributes, such as location, size, amenities, and transaction history.

At the core of the Transformer model is the Self-Attention Mechanism, which computes dependencies between features to enhance learning. The attention mechanism is mathematically defined as:

$$Attention = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{19}$$

where:

- Q (Query), K (Key), and V (Value) are linear projections of the input X, allowing the model to determine the relative importance of each feature in predicting property prices.
- d_k is the dimensionality of the key vectors, which helps scale the dot product for numerical stability.
- The $softmax$ function normalizes attention scores, ensuring that the model focuses on the most relevant features.

The Multi-Head Attention mechanism applies multiple attention operations in parallel, enabling the model to learn different aspects of feature relationships simultaneously. This enhances the model's ability to capture complex dependencies, such as the influence of neighborhood trends on property prices. To maintain stable training and prevent gradient vanishing, Residual Connections and Layer Normalization are applied after each attention block. These techniques help the model retain important information while allowing deeper architecture to converge efficiently. Following the attention mechanism, the output passes through a Feed-Forward Network (FFN), which further transforms the feature representation:

$$FFN(Z) = ReLU(ZW_1 + b_1)W_2 + b_2 \tag{20}$$

where:

- Z is the output from the attention layer.
- W_1 and W_2 are learned weight matrices, and b_1, b_2 are biases.
- The $ReLU$ (Rectified Linear Unit) activation function introduces non-linearity, helping the model capture complex interactions between features.

The final output is passed through dense layers to generate the predicted property price ŷ, minimizing the prediction error using the Mean Squared Error (MSE) loss function:

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2 \quad (21)$$

where:

- y_i is the actual price of the property,
- $\widehat{y_i}$ is the predicted price,
- N is the total number of property transactions.

The model's performance is assessed using key regression metrics, including R² (coefficient of determination), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). These metrics quantify the accuracy of predictions, enabling the model to effectively learn from historical transaction data and improve forecasting precision. By leveraging the Transformer architecture, the model efficiently captures non-linear interactions, spatial dependencies, and temporal trends, making it a robust tool for predicting real estate prices in Dubai's dynamic market.

3.3.7 Deep and Cross Network (DCN)

The Deep & Cross Network (DCN) is a deep learning architecture specifically designed to model complex feature interactions in structured, tabular data, making it highly suitable for the Dubai real estate transaction dataset. This model integrates two key components: a deep neural network (DNN) and a cross-network. The DNN captures non-linear relationships between features, while the cross-network explicitly models direct feature interactions, which are critical for predicting real estate prices accurately.

In the context of the Dubai real estate dataset, features like property size, location, number of bedrooms, transaction type, transaction date, and other property-specific features serve as input variables. The deep network learns complex patterns in these features, identifying non-linear relationships, while the cross-network focuses on direct interactions (e.g., how property size and location combine to influence price). By combining these two networks, the DCN model can effectively learn both high-level representations and explicit feature interactions, resulting in more accurate and interpretable predictions. For instance, it can reveal how specific combinations of features drive property prices in different market conditions. This makes DCN a robust solution for predicting Dubai real estate prices, leveraging the rich dataset to generate precise and reliable forecasts across varying market environments.

This combination of feature learning and interaction modeling helps create a highly predictive model that adapts to the complexity and variability of Dubai’s dynamic real estate market.

3.3.7.1 Deep Network

The deep component of the Deep & Cross Network (DCN) captures complex, non-linear patterns in the data by stacking multiple layers of neurons, each layer progressively learning higher-order feature interactions. In the context of the Dubai real estate transaction dataset, the deep network analyzes input features such as location, property size, number of bedrooms, transaction type, transaction date, and other property-specific attributes. By passing the data through several layers, the deep component learns to represent these features in high-dimensional space, uncovering intricate relationships that may not be immediately apparent in the raw data.

For instance, the network might learn that the impact of location on property prices varies significantly depending on the size of the property or the number of bedrooms. These kinds of non-linear relationships are difficult to capture with traditional linear models, but the deep network excels in recognizing these complex, multi-faceted interactions. Each neuron in a layer receives input from the previous layer and applies a non-linear activation function, allowing the network to detect subtle patterns that influence the target variable—in this case, the property price.

As the data is processed through each successive layer, the model builds increasingly sophisticated feature representations, helping it generalizes well to unseen data. This deep architecture allows the model to automatically learn the most relevant feature combinations without requiring explicit feature engineering. The ability to capture hidden patterns makes the deep network particularly powerful for real estate data, where numerous factors—such as market trends, property features, and transaction types—combine in complex ways to determine property values.

By learning these non-linear patterns, the deep component significantly enhances the model’s ability to provide accurate property price predictions across diverse market conditions and property types in Dubai. The deeper layers of the network can recognize latent factors affecting prices, such as market seasonality or buyer demand in specific locations, ultimately improving the model's overall predictive performance.

The deep network represented by,

$$h^{l+1} = f(W^l h^l + b) \qquad (22)$$

Where,

- h^l is the activation at layer l
- $W^l$and are the weights and biases of the i^th layer
- f (·) is the activation function.

3.3.7.2 Cross Network

The cross-network in the Deep & Cross Network (DCN) explicitly models feature interactions by combining the raw input features in a multiplicative manner across several layers. This design enables the model to capture direct, pairwise interactions between features, allowing it to learn how different features work together to influence the target variable, in this study, the property price. Unlike the deep component, which focuses on capturing complex, non-linear relationships, the cross-network is specifically designed to handle more straightforward feature interactions efficiently.

In the context of the Dubai real estate transaction dataset, the cross-network plays a critical role in modeling interactions between features such as property size, location, number of bedrooms, and transaction type. For example, the effect of property size on price might differ significantly depending on the location, a large property in a prime location will likely have a much higher price than a similarly sized property in a less desirable area. The cross-network can directly capture this type of interaction without requiring the deep network to learn it indirectly over multiple layers.

The cross terms in the network are formed by multiplying pairs (or groups) of features, creating a structure where these interactions are represented explicitly at each layer. For instance, in the Dubai dataset, the interaction between property size and transaction type (e.g., off-plan sales vs. secondary market transactions) could also play a crucial role in determining prices, as different transaction types might influence how much buyers are willing to pay for properties of varying sizes.

One of the strengths of the cross-network is its ability to handle low-dimensional, structured data like the tabular data found in real estate transactions. Unlike unstructured data, such as images or text, where deep layers are necessary to capture complex, hierarchical patterns, tabular data often benefits from explicit modeling of feature interactions. The cross-network efficiently captures these direct relationships, reducing the need for multiple layers and making it faster and easier for the model to learn meaningful interactions between features.

The cross network represented as,

$$x^{l+1} = x^0 * W^l x^l + b^l + x^l \quad (23)$$

Where,

- x^0 represents the original input features.
- W^l and b^l are the weights and biases of the cross-network at layer l.
- x^l is the output from the previous cross-layer.

3.3.7.3 Deep and Cross networks

The outputs of the deep and cross networks are combined to predict the property price, allowing the model to learn both high-level feature representations and explicit feature interactions simultaneously. This combination enhances prediction accuracy by capturing complex patterns and direct relationships between features.

The final prediction is,

$$\hat{y} = \sigma(W_{final}[h^L, x^L]b_{final}) \quad (24)$$

Where,

- $\hat{y}$ is the predicted property price.
- σ (·) is the activation function for the output layer (e.g., linear for regression tasks).
- h^L is the output of the deep network and x^L is the output of the cross-network.
- W_{final} and b_{final} are the weights and biases for the final prediction layer.

3.3.8 TabTransformer

The TabTransformer is a deep learning model designed specifically for tabular datasets, like the Dubai real estate transaction dataset. It integrates a transformer-based architecture to capture meaningful feature interactions while handling categorical and numerical data. A study highlights the need for robust network intrusion detection systems (NIDS) amidst rising cyber threats. It found that TabTransformer outperformed traditional models like SVM, LR, MLP, and a Voting Model in precision, recall, and F1-score. Its ability to handle both categorical and numerical features, along with capturing complex patterns in tabular data, makes it a powerful tool for real-time cyber threat detection (Wang et al., 2024).

Important categorical features such as location and transaction type are transformed using embeddings, which convert these discrete values into continuous vectors. This process captures the underlying relationships between categories, allowing the model to treat features like different locations or transaction types as continuous variables, making it easier to identify patterns and trends. For example, similar neighborhoods or transaction types may have similar embeddings, enabling the model to generalize better across comparable properties. Let $x_i$represent a categorical feature, and $E(x_i)$ its corresponding embedding. This transform x_i into a dense vector z_i.

$$z_i = E(x_i) \tag{25}$$

The self-attention mechanism in TabTransformer plays a pivotal role in the model's performance by dynamically focusing on the most relevant features and feature interactions at each prediction step. Unlike traditional models, where all features are treated equally, the attention mechanism gives more weight to those features that are most influential for predicting the target variable, in this case, property price. This is particularly useful in real estate data, where factors like location and property size may have different levels of importance depending on the specific property or transaction.

For instance, in a high-demand neighborhood, the location feature may carry more weight in the prediction, while for properties in less sought-after areas, other features like property size or number of bedrooms might be more significant. By dynamically adjusting the importance of each feature, the self-attention mechanism enables the model to capture complex relationships and interactions that are critical in a diverse and dynamic market like Dubai. The attention score for feature x_i in relation to other features x_j is given by,

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{26}$$

Where,

- Q is the query matrix.
- K is the key matrix.
- V is the value matrix.

- d_k is the dimension of the key vectors.

At the same time, numerical features such as property size, number of bedrooms, and transaction date are processed through normalization, which scales these values to a common range. This step is critical to ensure that numerical features are on a comparable scale, preventing any one feature from dominating the model simply due to its magnitude. Normalization also helps to stabilize and accelerate the model's convergence during training, ensuring more efficient learning from the data. If x_n represents a numerical feature, it can be scaled as follows.

$$Z_n = (\frac{x_n - \mu}{\sigma}) \quad (27)$$

Where,

- $x_n$represent the numerical feature.

TabTransformer applies the attention mechanism across all feature embeddings (categorical and numerical) over multiple layers, learning complex dependencies between the features. This is especially useful in a dataset like Dubai real estate, where interactions between variables like location, size, and transaction type are crucial for accurate predictions. After the attention layers, the output embeddings are passed through fully connected layers to predict the target variable, property price. The model minimizes a loss function (e.g., mean squared error, L(y,$\hat{y}$) to improve prediction accuracy,

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2 \quad (28)$$

Where,

- $\hat{y_i}$ is the predicted price.
- y_i is the actual price.

3.3.9 Feature-Token Transformer (FT-Transformer)

The Feature-Token Transformer (FT-Transformer) is a deep learning architecture designed to handle tabular data effectively, such as the Dubai real estate transaction dataset. It adapts the power of transformer models, originally designed for natural language processing, to model tabular datasets by learning the interactions between individual features and their values. Here's how it works on this dataset. For each feature in the dataset (e.g., location, transaction type, property size, number of bedrooms), the FT-Transformer first converts both categorical features and numerical features into embeddings or continuous dense vectors. This allows the model to capture both the underlying relationships between categorical values and scale numerical features appropriately. For example, let x_i be a feature from the dataset, and $E(x_i)$ be its corresponding embedding. The feature embedding transforms x_i into a dense vector z_i, which is then used by the transformer layers.

$$z_i = E(x_i) \tag{29}$$

Biochar adsorbents from food and agricultural wastes are widely used to remove heavy metals from wastewater. However, biochar's varied properties and experimental conditions make estimating adsorption capacity (qe) difficult. To address this, machine learning (ML) and three deep learning (DL) models were developed using 1,518 data points. The recursive feature elimination method identified 14 key inputs from 28. The FT-transformer model outperformed other ML and DL models, achieving the highest test R^2 (0.98) and lowest RMSE (0.296) and MAE (0.145)(Jaffari et al., 2024).

The FT-Transformer uses a multi-head self-attention mechanism, similar to the classic transformer architecture, to learn the relationships between different features. The self-attention mechanism models how different features influence each other dynamically and weighs the importance of each feature for the prediction task. The attention mechanism computes a weighted sum of the embedded feature tokens. The attention score between two features x_i and x_j is calculated using the following formula:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{30}$$

Where,

- Q is the query matrix, representing the current feature of focus.
- K is the key matrix, representing all features.
- V is the value matrix, representing the actual feature values.
- d_k is the dimension of the key vectors.

This mechanism allows the model to focus more on important features (e.g., location or property size) that have a greater impact on the target variable (property price).

After embedding and self-attention, the model captures both linear and non-linear interactions between features. For instance, in the Dubai real estate dataset, it can learn how the interaction between features like location and property size affects the target variable (property price). The attention mechanism helps to capture these interactions more effectively than traditional machine learning models.

After multiple layers of self-attention, the output embeddings are fed into fully connected layers to predict the target variable, in this case, property price. The model minimizes a loss function, typically mean squared error (MSE), during training.

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2 \tag{31}$$

Where,

- y_i is the actual property price.
- $\widehat{y_i}$ is the predicted property price.
- N is the number of data points.

3.3.10 Gated Recurrent Unit (GRU)

The Gated Recurrent Unit (GRU) is a type of recurrent neural network (RNN) designed to handle sequential data efficiently. While typically used for time-series data or tasks like natural language processing, GRU can also be adapted for tabular datasets, such as the Dubai real estate transaction dataset. In this context, GRU can help capture temporal dependencies, such as how past property transactions influence future prices or trends in the market. A study developed a unified framework for implementing advanced deep learning models—LSTM, GRU, CNN, and their hybrids—to predict the next day's closing price of the S&P500-60 real estate index. It was found that the base GRU model, followed by the bidirectional GRU, provided the most accurate predictions for the index's closing price (Rimal et al., 2024).

Here’s how the GRU model would work on the Dubai real estate dataset. In the Dubai real estate dataset, certain features such as the transaction date or historical property prices are time-dependent, meaning past transactions could influence future prices. GRU is ideal for modeling such dependencies over time, as it can remember previous information and use it to inform future predictions.

For example, sequential features like historical transaction dates and prices can be used to predict future property prices. Other non-sequential features (like location, number of bedrooms, etc.) can be fed as static inputs along with the time-dependent features.

GRU simplifies the traditional RNN by using gates to control the flow of information, which helps in capturing long-term dependencies more effectively. A GRU cell consists of two gates, Update gate and Reset gate

3.3.10.1 Update Gate

Determines how much of the previous information should be passed to the current time step. The update gate plays a key role in maintaining the memory of the GRU. Since sequential data, such as historical property prices or transaction dates, often contains relevant patterns over time, the model needs to decide which past information is still useful for the prediction at the current step. For instance, in real estate, past trends in property prices may influence future price predictions, but certain information may become less relevant as the market evolves.

If the update gate outputs a value close to 1, it allows the GRU to retain most of the past information (from the previous hidden state). This is useful when the past context remains important for the current decision.

If the update gate outputs a value close to 0, the GRU will focus more on the new information coming from the current time step, ignoring most of the previous state.

By adjusting the balance between past and new information, the update gate ensures that the model keeps track of important trends while adapting to new data. In the case of the Dubai real estate transaction dataset, this could mean keeping a memory of past property prices and trends while also adapting to recent market shifts.

Mathematical representation of update gate is,

$$z_t = \sigma(W_z * [h_{t-1}, x_t] + b_z) \tag{32}$$

Where,

- z_t is the update gate at time step $t.$
- W_z is the weight matrix for the update gate.
- h_{t-1} is the hidden state from the previous time step (representing the past information),

- x_t is the current input (new information),
- b_z is the bias term,
- σ is the sigmoid function which ensures the update gate outputs a value between 0 and 1.

3.3.10.2 Reset Gate

Controls how much of the past information should be "forgotten.". The reset gate plays a critical role in controlling the influence of past information on the current step. In sequential tasks, like predicting future property prices based on historical data in the Dubai real estate transaction dataset, some aspects of past information may become irrelevant over time. For instance, older transactions from years ago may no longer have a strong influence on current market conditions. The reset gate helps the GRU model "forget" such outdated or irrelevant information, allowing the model to focus more on recent trends or new data that are more relevant to the current prediction.

If the reset gate outputs a value close to 0, its "resets" the hidden state by largely ignoring the past information. This is useful when the current input data is significantly different or more important than the historical data. If the reset gate outputs a value close to 1, it allows the model to retain most of the past hidden state, meaning that the past context is still important and should influence the current decision.

This gate ensures that the GRU model can dynamically adjust how much it relies on past information, depending on the context of the input data. In the Dubai real estate dataset, for example, the reset gate might decide to forget older property price trends when more recent data shows a rapid shift in market conditions, allowing the model to adapt quickly to these changes.

The mathematical representation of reset gate,

$$r_t = \sigma(W_r * [h_{t-1}, x_t] + b_r \qquad (33)$$

Where,

- r_t is the reset gate at time step t.
- W_r is the weight matrix for the reset gate,
- h_{t-1} is the hidden state from the previous time step (representing past information),
- x_t is the current input (new information),

- b_r is the bias term,
- σ is the sigmoid function that produces a value between 0 and 1, determining how much of the previous hidden state to keep.

The reset gate controls how much of the past information to consider in calculating the current memory content.

$$\widehat{h_t} = \tan h\,(W * \left[r_{t\odot h_{t-1}}, x_t\right] + b) \tag{34}$$

Where,

- $\widehat{h_t}$ is the candidate hidden state (new information).
- r_t is the reset gate applied elementwise ($\odot$).
- W is the weight matrix for the new information.
- $\tan h$ is the hyperbolic tangent activation function.
- x_t is the current input.

The final hidden state is a combination of the previous hidden state and the candidate hidden state, controlled by the update.

$$h_t = (1 - z_t)\odot h_{t-1} + z_t\odot\widehat{h_t} \tag{35}$$

Where,

- h_t is the final hidden state at time step t,
- z_t is the update gate controlling how much of the previous hidden state (h_{t-1}) and the new candidate state ($\widehat{h_t}$) should be used.

For the Dubai real estate dataset, GRU can model temporal trends in property prices over time by processing sequential data like the transaction dates and property prices. The GRU takes the property features (e.g., location, size, number of bedrooms) along with the time-sequenced transaction data (e.g., historical transaction prices, dates) as input. At each time step, the GRU updates its hidden state using the gates to selectively retain or discard information from the previous time steps, ensuring that it learns the temporal dependencies. Based on the sequence of transactions and property attributes, GRU can predict the future property prices or identify trends in the real estate market. The model minimizes a loss function, such as mean squared error (MSE), to optimize the accuracy of the prediction.

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2 \quad (36)$$

Where,

- y_i is the actual property price.
- $\widehat{y_i}$ is the predicted price from the GRU model.

3.4 Model Evaluation

Model evaluation is a key aspect of assessing the performance of machine learning (ML) and deep learning (DL) models built on the Dubai real estate transaction dataset, available from Dubai Pulse. After training the models, various evaluation metrics are used to measure how well the predictions match actual property prices. Common metrics include:

3.4.1 Root Mean Squared Error (RMSE): RMSE measures the square root of the average squared differences between predicted and actual property prices. It gives more weight to larger errors, making it particularly sensitive to significant deviations in predictions. RMSE is calculated as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{1=1}^{n}(y_i - \hat{y}_i)^2} \quad (37)$$

where

- y_i is the actual property price.
- $\widehat{y_i}$ is the predicted price.
- n is the number of predictions.

3.4.2 Mean Absolute Error (MAE): MAE measures the average absolute differences between the predicted and actual prices, making it a straightforward metric for understanding prediction accuracy. Unlike RMSE, MAE gives equal weight to all errors and is less sensitive to outliers. The formula for MAE is:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}| \quad (38)$$

where

- y_i is the actual property price.
- $\widehat{y_i}$ is the predicted price.

- n is the number of predictions.

3.4.3 R-Squared (R²): R-Squared indicates the proportion of variance in property prices that can be explained by the model. It ranges from 0 to 1, with values closer to 1 indicating better model performance. An R^2 of 1 means the model perfectly predicts the property prices, while an R^2 of 0 suggests that the model does no better than a simple mean prediction. The R^2 formula is:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2)}{\sum_{i=1}^{n}(y_i - y_{mean})^2} \tag{39}$$

where y_{mean} is the mean of the actual prices.

These metrics help in evaluating the accuracy, reliability, and overall effectiveness of the models developed for predicting Dubai real estate prices, ensuring that they can generalize well to unseen data.

CHAPTER 4

IMPLEMENTATION AND ANALYSIS

4.1 Introduction

This section explains how the methodology was applied to the Dubai Transactions dataset, covering each step from preparing the data to evaluating the models. It describes how missing values were handled, how categorical data was converted into a usable format, and how numerical features were scaled. It also explains why certain machine learning and deep learning models were chosen and how their performance was improved through tuning. By comparing the models using key evaluation measures, this section helps identify the best approach for predicting real estate prices in Dubai.

4.2 Data Preprocessing

Data preprocessing begins with loading the Dubai Land Department (DLD) Transactions Open Dataset, which consists of 1,339,796 rows and 46 columns. To ensure efficient processing while maintaining representativeness, a random 5%, which is 66990 records, sample is selected, balancing computational efficiency with data integrity. The dataset contains categorical columns described in both English and Arabic; to simplify analysis, Arabic columns are excluded. The next step involves filtering and retaining 13 key columns that significantly impact real estate price prediction. These include property attributes such as type, sub-type, number of rooms, and parking availability, alongside transaction-related details like sale price per square meter, procedure area, and transaction group. Registration and usage types are considered for regulatory insights, while the number of parties involved in different roles provides an understanding of transaction complexity. The target variable, actual worth, represents the final property price to be predicted. By carefully selecting relevant features and optimizing the dataset, unnecessary information is removed, noise is reduced, and predictive modeling becomes more efficient and accurate.

Table 1: Overview of Selected Features

Categorical Variables	property_type_en, property_sub_type_en, reg_type_en, property_usage_en, rooms_en, trans_group_en
Numerical variables	Has_parking, meter_sale_price, no_of_parties_role_1, no_of_parties_role_2, no_of_parties_role_3, procedure_area, actual_worth

From the above table we could see that it classifies the selected features into **categorical** and **numerical** variables, both of which play a crucial role in real estate price prediction.

Categorical Variables:

These variables describe property and transaction characteristics in a qualitative manner, often influencing price through non-numerical factors.

- **property_type_en**: Defines the general category of the property (e.g., apartment, villa, or commercial).
- **property_sub_type_en**: Provides a more specific classification within the property type (e.g. apartment, shop, or office).
- **reg_type_en**: Specifies the type of property registration, such as off plan or existing property.
- **property_usage_en**: Indicates whether the property is designated for residential, commercial, or mixed-use purposes.
- **rooms_en**: Represents the total number of rooms in the property, a key factor in determining property value.
- **trans_group_en**: Categorizes the type of transaction, such as first-time sales, resales, or mortgage transactions.

Numerical Variables:

These variables represent measurable data points that directly impact property valuation.

- **has_parking**: A binary indicator (0 or 1) showing whether the property includes parking.
- **meter_sale_price**: The price per square meter, providing a standardized measure of property cost.
- **no_of_parties_role_1, no_of_parties_role_2, no_of_parties_role_3**: Represent the number of parties involved in different roles in the transaction, affecting its complexity.
- **procedure_area**: Indicates the total area of the property in square meters, a key determinant of price.

- **actual_worth**: The final sale price of the property, which serves as the target variable for predictive modeling.

By distinguishing between categorical and numerical variables, the dataset enables efficient preprocessing, such as encoding categorical features and normalizing numerical ones, leading to improved accuracy in predictive modeling.

4.3 Data Cleaning

Numerical features such as meter_sale_price, actual_worth, and procedure_area contain outliers that can negatively impact the performance of predictive models by skewing the distribution and introducing bias. To address this, we use the Interquartile Range (IQR) method, a widely used statistical technique for detecting and handling outliers. The IQR method identifies extreme values by calculating the first quartile (Q1) and third quartile (Q3) and determining the IQR as Q3 - Q1. Any data points falling below Q1 - 1.5 * IQR or above Q3 + 1.5 * IQR are considered outliers. These outliers can be removed to improve model accuracy and stability. By applying this approach, we ensure that the dataset remains robust and reflective of realistic property prices, preventing extreme values from distorting the predictions.

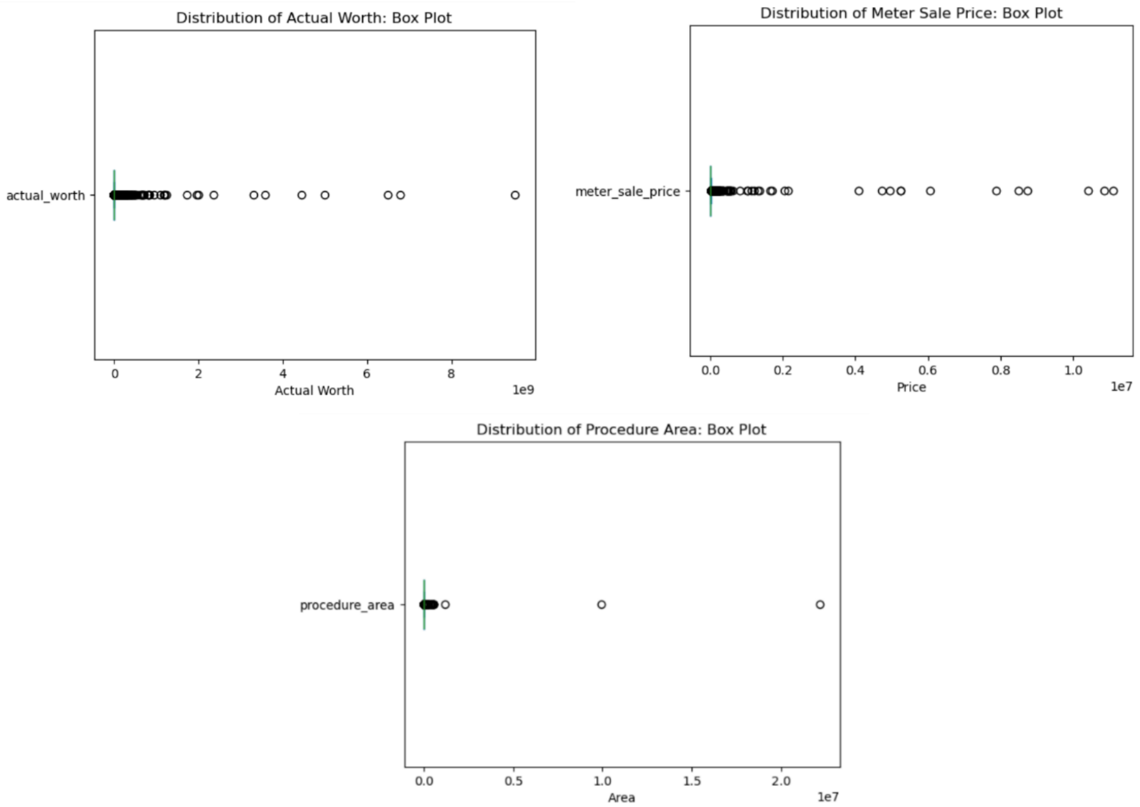


Fig 2. Numerical Features before handling outliers

The box plots for actual_worth, meter_sale_price, and procedure_area reveal significant outliers, indicating a highly skewed distribution in the dataset. Most values are concentrated within a lower range, while extreme values extend far beyond the typical spread, suggesting the presence of high-end properties or exceptionally large land plots. These outliers can distort statistical analysis and predictive modeling, leading to biased estimations.

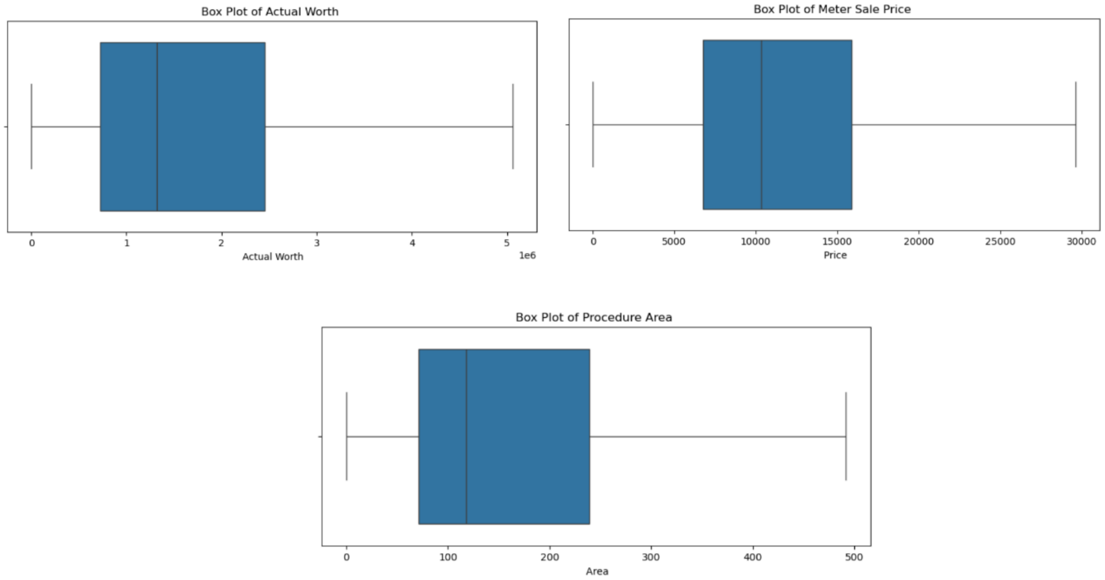


Fig 3. Numerical Features After Handling outliers

After handling outliers, the box plots for Meter Sale Price, Procedure Area, and Actual Worth appear more compressed and evenly distributed, indicating that extreme values have been effectively managed. The whiskers now extend over a reasonable range, reducing skewness and ensuring a more representative distribution of property prices and areas. This adjustment minimizes the influence of extreme values, leading to a dataset that better reflects typical market trends.

For categorical values, 24% of the dataset's rows contain at least one missing value, primarily in rooms_en and property_sub_type_en. Since 96% of these missing values occur in both columns simultaneously, dropping all affected rows would lead to excessive data loss. Instead, missing values in these columns are replaced with 'N/A' to retain potential relationships with the target variable. The remaining 4% of missing values in other columns are removed to ensure data consistency.

With this approach, the sample set with selected features is now clean, preserving data integrity, minimizing loss, and ensuring a reliable dataset for further analysis or modeling.

4.4 Data Analysis

We are performing Exploratory Data Analysis (EDA) to uncover patterns, distributions, and relationships within the 13 selected features that will be used for building our real estate price prediction model. EDA helps us understand the data structure, identify outliers, handle missing values, and detect trends that influence property prices. By analyzing histograms and statistical summaries, we observe skewness in numerical features, variations in categorical distributions, and potential correlations among variables. This process allows us to uncover hidden patterns, detect anomalies, and validate feature importance, ensuring that only relevant attributes contribute to model performance. The insights gained from EDA will guide feature engineering, data preprocessing, and model optimization for accurate real estate price predictions.

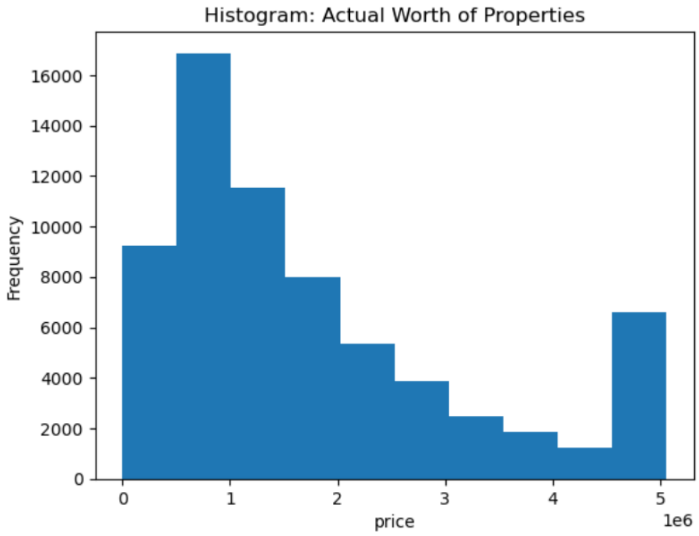


Fig 4. Histogram for Actual Worth

The actual worth of properties (Fig 4.) exhibits a right-skewed distribution, indicating that most properties are valued on the lower end of the scale. A significant portion of transactions falls below 2 million, while higher-priced properties above 4 million are comparatively fewer. However, a secondary peak around 5 million suggests the presence of a distinct cluster of high-end properties, likely representing luxury real estate or premium developments. This distribution highlights the disparity in property values, with most transactions occurring in the more affordable range while a smaller segment caters to the high-end market.

The procedure area of properties (Fig 5.) exhibits a right-skewed distribution, with most properties having a smaller area. Most properties are concentrated within 0 to 150 square meters, while the number of properties gradually decreases as the area increases beyond 300 square meters. A notable peak near 500 square meters suggests the presence of some larger properties in the dataset, likely representing spacious villas or high-end real estate.

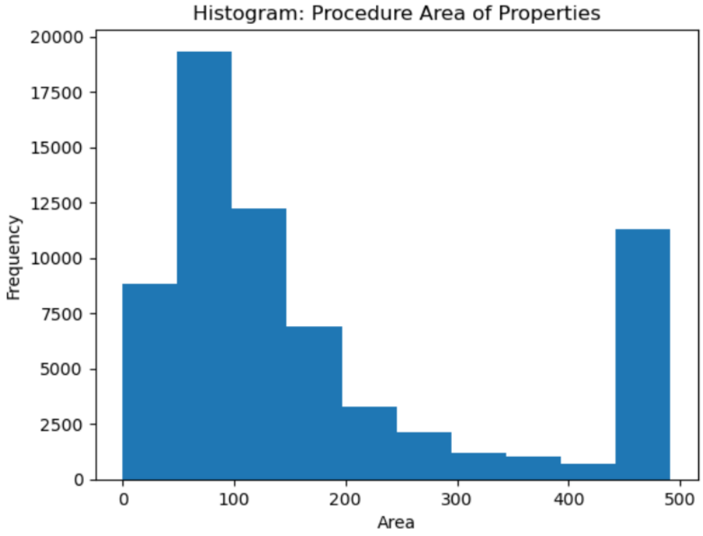


Fig 5. Histogram for Procedure Area

The dataset is predominantly composed of units, accounting for 69.4% of the properties, highlighting the dominance of apartments or similar residential structures in the real estate market. Villas represent 19.5%, reflecting a substantial share of standalone homes. Land transactions make up 8.4%, indicating a smaller but notable market segment for undeveloped or investment plots. The remaining 2.6% are likely to include buildings or other specialized property types, such as commercial spaces.

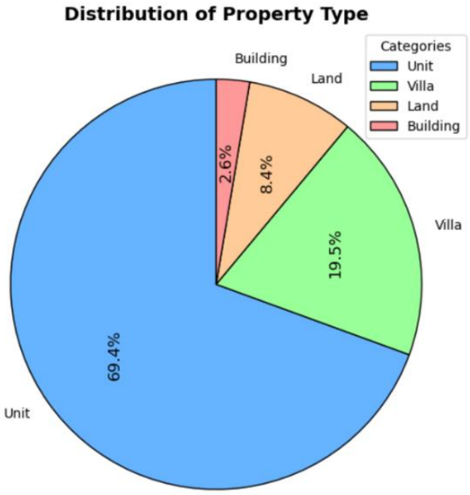


Fig 6. Distribution of Property Type

From the figure below, it is evident that 68.9% of registrations are for existing properties, while 31.1% are for off-plan properties.

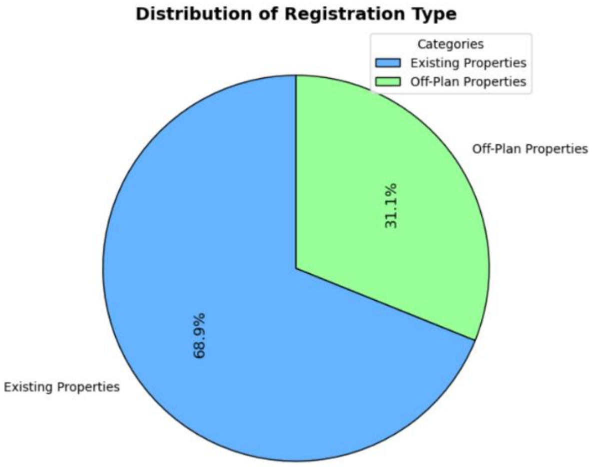


Fig 7. Distribution of Registration type

Figure 8 presents a scatter plot depicting the relationship between Procedure Area and Actual Worth, revealing a positive correlation where larger areas generally correspond to higher property values.

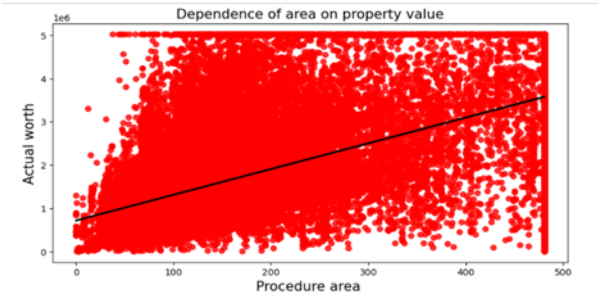


Fig.8 Relationship between area and actual worth

However, the significant variance indicates that area alone is not a strong predictor, as other factors also influence property prices.

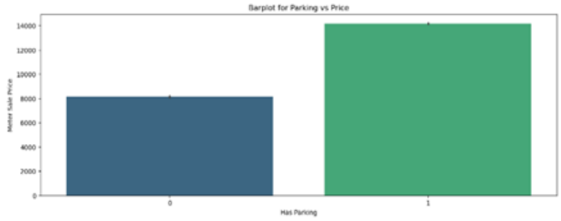


Fig.9 Relationship between properties with parking and sale price

Figure 9 highlights that the availability of parking is a significant factor influencing property prices.

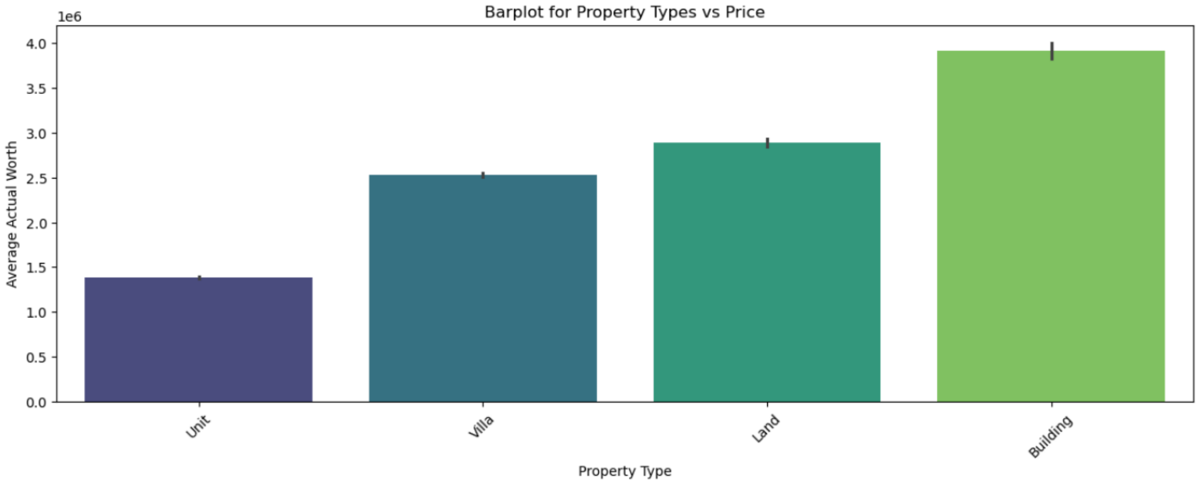


Fig 10. Relationship between Property type and Actual Worth

The plot shows that Buildings have the highest average worth, followed by Land, Villas, and Units, respectively. This suggests that larger property types, such as Buildings and Land, tend to have higher market values compared to smaller property types like Units.

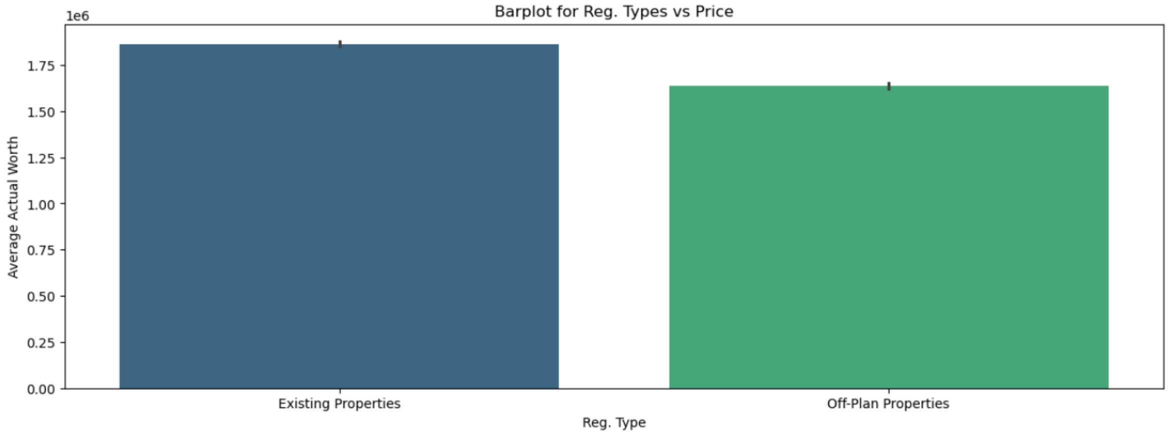


Fig 11. Relationship between Registration type and Actual Worth

And we could see that existing properties are more expensive than off-plan properties.

4.5 Model Design

The process begins with preparing the dataset for machine learning models by converting categorical variables into numerical representations using one-hot encoding. The target variable, actual_worth, is separated from the features, and all numerical features are normalized using StandardScaler to ensure consistency in scale. To enhance model performance, SelectKBest with f_regression is applied to select the top 10 most relevant features. Finally, this sampled data is split into training and testing sets, with 30% of the sampled data reserved for testing, ensuring a balanced evaluation of the model's predictive accuracy.

For deep learning models, the process starts with handling categorical data by identifying key features such as property type, registration type, property usage, and number of rooms. These categorical variables are converted into numerical representations using Label Encoding, ensuring they are suitable for neural network architectures. The dataset is then split into features (X) and the target variable, actual_worth (y), followed by a train-test split, where 80% of the data is allocated for training and 20% for testing, maintaining consistency with a fixed random state. Additionally, numerical features like meter sale price, procedure area, and the number of parties involved are identified separately to facilitate normalization and feature scaling, which are crucial for optimizing deep learning model performance.

The core difference is that deep learning models require extra preprocessing steps like normalization and feature scaling to enhance performance, while traditional ML models can work directly with encoded categorical and raw numerical data without additional transformations.

The next section will explain entire model architecture for each model. This process involves optimizing the parameters that control the behavior of machine learning algorithms and Deep Learning algorithms to improve model performance. By selecting the best combination of hyperparameters, we aim to achieve better accuracy, reduce overfitting, and enhance the generalization ability of each model.

4.5.1 Support Vector Regression (SVR)

A Support Vector Regression (SVR) model is defined and set up for hyperparameter tuning using RandomizedSearchCV. The goal is to find the optimal set of hyperparameters that improve the model's performance. An extended hyperparameter grid is specified, including various kernel types (linear, rbf, and poly), a broader range of regularization parameters (C values from 0.1 to 1000), tolerance levels (epsilon), kernel coefficient options (gamma), and the degree of the polynomial kernel. This wide range of parameters allows the model to explore different configurations and identify the best combination. The RandomizedSearchCV method is then used to randomly sample from this grid, performing 15 iterations of parameter combinations while employing 3-fold cross-validation to evaluate model performance. The model is evaluated using the R^2 score, a measure of how well the model fits the data, and all available computational cores are utilized (n_jobs=-1) to speed up the process. After fitting the model on the training data, it seeks the best hyperparameters, which are then used to optimize the SVR model for prediction tasks.

4.5.2 Random Forest (RF)

A RandomForestRegressor model is defined and prepared for hyperparameter tuning using RandomizedSearchCV. A grid of potential hyperparameters for the Random Forest is specified, which includes the number of trees in the forest (n_estimators), the maximum depth of the trees (max_depth), the minimum number of samples required to split a node (min_samples_split), the minimum number of samples needed at each leaf node (min_samples_leaf), and whether or not to use bootstrapping for sampling (bootstrap). These parameters are tuned to improve the model's accuracy and prevent overfitting. RandomizedSearchCV is used to randomly sample from this grid and evaluate combinations across 10 iterations using 3-fold cross-validation. The performance is measured using the R^2 score, which indicates how well the model fits the data. To speed up the search, all available computational cores are utilized (n_jobs=-1). The model is then fitted on the training data, searching for the best combination of hyperparameters to optimize the Random Forest model for accurate predictions. The repeated fit statement (fit (X_train, y_train)) seems redundant and could be cleaned up.

4.5.3 Gradient Boosting Regression (GBR)

A Gradient Boosting Regressor (GBR) model is being set up for hyperparameter tuning using RandomizedSearchCV. A range of hyperparameters is specified to optimize the model's performance. These parameters include the number of boosting stages (n_estimators), the step size shrinkage (learning_rate), the maximum depth of each tree (max_depth), the minimum number of samples required to split a node (min_samples_split), the minimum samples at each leaf node (min_samples_leaf), and the fraction of samples used for fitting each base learner (subsample). By tuning these hyperparameters, the model can improve accuracy, prevent overfitting, and achieve better generalization. RandomizedSearchCV randomly selects combinations from this grid and evaluates them through 10 iterations using 3-fold cross-validation. Model performance is measured using the R^2 score, and all available computational cores are used (n_jobs=-1) to speed up the tuning process. Finally, the model is trained using the optimized parameters on the training data to achieve the best possible performance.

4.5.4 Extreme Gradient Boost (XGBoost)

An XGBoost Regressor model is optimized using RandomizedSearchCV to find the best set of hyperparameters. The search space includes key parameters such as the number of trees (n_estimators), learning rate (learning_rate), and maximum tree depth (max_depth), which control the model's complexity and learning process. Additional parameters like the fraction of samples used for training (subsample) and the fraction of features used per tree (colsample_bytree) help regulate model variance and prevent overfitting. The tuning also considers min_child_weight, which determines the minimum sum of instance weights needed in a child node, gamma, which controls the minimum loss reduction required for a split, and regularization terms (reg_alpha for L1 and reg_lambda for L2) to improve generalization. RandomizedSearchCV runs 10 iterations with different hyperparameter combinations, performing 3-fold cross-validation to evaluate model performance based on the R^2 score. The tuning process utilizes all available computational cores (n_jobs=-1) to enhance efficiency. Once the best hyperparameters are identified, the model is trained on the dataset to achieve optimal predictive performance.

4.5.5 Ridge Regression and Lasso Regression

A Ridge Regression model is optimized using RandomizedSearchCV to find the best regularization strength. The hyperparameter grid consists of different values for alpha, which controls the amount of regularization applied to the model. A logarithmic scale (np.logspace(-4, 4, 50)) is used to explore a wide range of values, ensuring that both small and large regularization strengths are considered. Ridge regression helps prevent overfitting by adding an L2 penalty to the model's coefficients. RandomizedSearchCV runs 10 iterations, selecting different alpha values and evaluating the model's performance using 3-fold cross-validation based on the R^2 score. The tuning process leverages all available computational cores (n_jobs=-1) to speed up execution. Once the best alpha value is determined, the model is trained using the optimal regularization strength to enhance predictive accuracy while maintaining generalization.

A Lasso Regression model is optimized using RandomizedSearchCV to identify the best regularization strength. The hyperparameter grid includes different values for alpha, which controls the level of L1 regularization applied to the model. A logarithmic scale (np.logspace(-4, 4, 50)) is used to explore a broad range of alpha values, ensuring that both weak and strong regularization effects are considered. Lasso regression is particularly useful for feature selection, as it can shrink some coefficients to zero, effectively removing less important features. RandomizedSearchCV is set to evaluate 10 different alpha values through 3-fold cross-validation, measuring performance using the R² score. To accelerate the tuning process, all available computational cores are used (n_jobs=-1). Once the optimal alpha is determined, the Lasso model is trained using this best parameter, balancing predictive accuracy while promoting sparsity in the feature set.

4.5.6 Transformer

Transformer models implemented by starting a Transformer Encoder Block is defined, which applies multi-head attention and feed-forward layers with dropout and layer normalization to enhance feature interactions. The model processes numerical features as direct inputs, while categorical features are embedded into dense vectors before being flattened. All features are then concatenated and reshaped to fit the Transformer structure. The encoded output passes through fully connected layers before producing the final regression output. The model is compiled using the Adam optimizer with a mean squared error (MSE) loss function, and it is trained for 80 epochs with a batch size of 256. A helper function prepares the input data for training by ensuring the correct format for both categorical and numerical features. The model leverages transformer attention mechanisms to capture complex relationships, making it well-suited for structured real estate data.

4.5.7 TabTransformer

TabTransformer-based deep learning model for regression, using attention mechanisms to enhance feature interactions. The model processes categorical features by embedding them into dense representations, while numerical features are reshaped to align with the embedded categorical inputs. These features are then concatenated and passed through a TabTransformer Encoder Block, which applies multi-head attention and feed-forward layers with dropout and layer normalization to capture complex dependencies. The transformed features are flattened and passed through fully connected layers, leading to the final regression output. The model is compiled using the Adam optimizer with mean squared error (MSE) loss and is trained for 80 epochs with a batch size of 256. A helper function ensures the correct formatting of inputs before training. By combining embedding-based feature representation with attention mechanisms, the TabTransformer model effectively learns patterns in structured real estate data.

4.5.8 Feature-Token Transformer (FT-Transformer)

This model uses Transformer architecture to process both categorical and numerical features for regression tasks. Categorical inputs are embedded, while numerical features are directly incorporated. These are concatenated, reshaped, and passed through an attention-based encoder with multi-head attention, dropout, and normalization to enhance feature interactions. A positional encoding layer further refines learning before the transformed data moves through dense layers for prediction. Optimized using Adam with MSE loss, it undergoes 80 training epochs with a batch size of 256 to improve predictive accuracy.FT-Transformer fully tokenizes all features, optimizing for tabular data, while the standard Transformer still distinguishes between numerical and categorical inputs.

4.5.9 Gated Recurrent Unit (GRU)

The GRU (Gated Recurrent Unit) processes structured data by embedding categorical features and directly using numerical features. These are concatenated and reshaped to introduce a sequential aspect before passing through the GRU layer, which captures temporal dependencies within the dataset. The GRU output is batch-normalized and fed into fully connected layers for further feature extraction, leading to the final regression output. The model is optimized using Adam and trained for 100 epochs with a batch size of 256, ensuring efficient learning from real estate data.

4.5.10 Deep and Cross Network (DCN)

The DCN (Deep & Cross Network) processes numerical and categorical features, embedding categorical ones and concatenating them with numerical inputs. The model has two parts: a deep part using dense layers for feature representations and a cross part that learns feature interactions via a custom CrossLayer. The outputs from both parts are combined and passed through a final dense layer for regression. The model is trained for 100 epochs with the Adam optimizer, MSE loss, and MAE evaluation, capturing complex feature interactions in the dataset.

The choice of epoch size for GRU, DCN, Transformer, FT Transformer, and TabTransformer varies based on model complexity. GRU needs fewer epochs due to its simpler, efficient architecture for sequential data. DCN requires more epochs to optimize its deep and cross layers. Transformer and TabTransformer converge in around 80 epochs, as their attention mechanisms enable faster learning. FT Transformer, being more complex, needs 100 epochs for optimal feature interaction. More complex models generally require more epochs to converge, while simpler ones like GRU need fewer.

4.6 Summary

The dataset analysis reveals that 69.4% of properties are units (apartments), 19.5% are villas, and 8.4% are land, with a smaller portion (2.6%) consisting of buildings or other property types. Existing properties dominate registrations at 68.9%, while 31.1% are off plan. Larger properties like buildings and land typically have higher market values than smaller units, and parking availability significantly influences property prices. Data preprocessing involves encoding categorical variables, normalizing numerical features, and selecting relevant features using SelectKBest. Machine learning models such as Support Vector Regression (SVR), Random Forest (RF), Gradient Boosting Regression (GBR), and XGBoost are tuned using RandomizedSearchCV for optimal hyperparameters to improve predictive performance. Regression models like Ridge and Lasso are optimized for regularization strength, while deep learning models like Transformer, TabTransformer, FT-Transformer, GRU, and Deep & Cross Network (DCN) leverage attention mechanisms and sequential data processing to enhance prediction accuracy. Training varies, with simpler models like GRU requiring fewer epochs and complex models like FT-Transformer needing more epochs for convergence. The overall goal is to optimize the models for better accuracy and generalization, avoiding overfitting.

CHAPTER 5

RESULTS AND EVALUATION

5.1 Introduction

The results and discussion section examines the performance of various machine learning and deep learning models in predicting real estate prices, analyzing key insights derived from the dataset. By comparing Machine Models such as XGBoost, Random Forest, and Ridge Regression with Deep Learning architectures like Transformers and GRU, this section evaluates their effectiveness in capturing complex market trends. Additionally, the impact of key features, including property type, procedure area, and parking availability, on actual worth is explored, providing a deeper understanding of price determinants. Ultimately, these findings contribute to optimizing predictive models for Dubai's real estate market, offering insights into the strengths and limitations of different approaches.

5.2 Univariate Analysis

Univariate analysis was conducted to examine the distribution and characteristics of individual variables within the dataset.

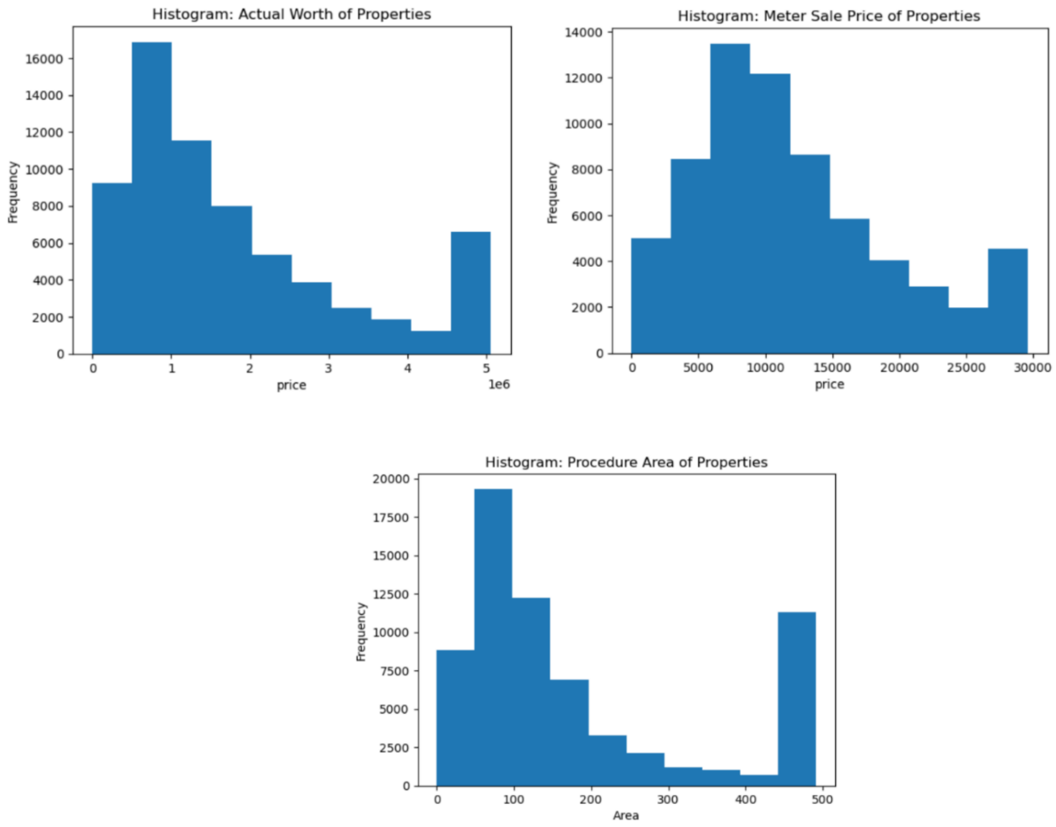


Fig 12. Histogram for Actual Worth, Meter Sale Price and Procedure Area

The distributions of Actual Worth, Meter Sale Price, and Procedure Area are right-skewed, indicating most properties fall within lower price and size ranges, with fewer high-end or larger properties. Actual Worth is concentrated below 2 million, with a secondary peak around 5 million. Meter Sale Price mostly falls under 10,000 per square meter, with fewer exceeding 20,000. The Procedure Area is mainly 0-150 square meters, with a peak near 500 square meters. These trends highlight affordability as a key factor, while secondary peaks suggest a distinct market for premium properties.

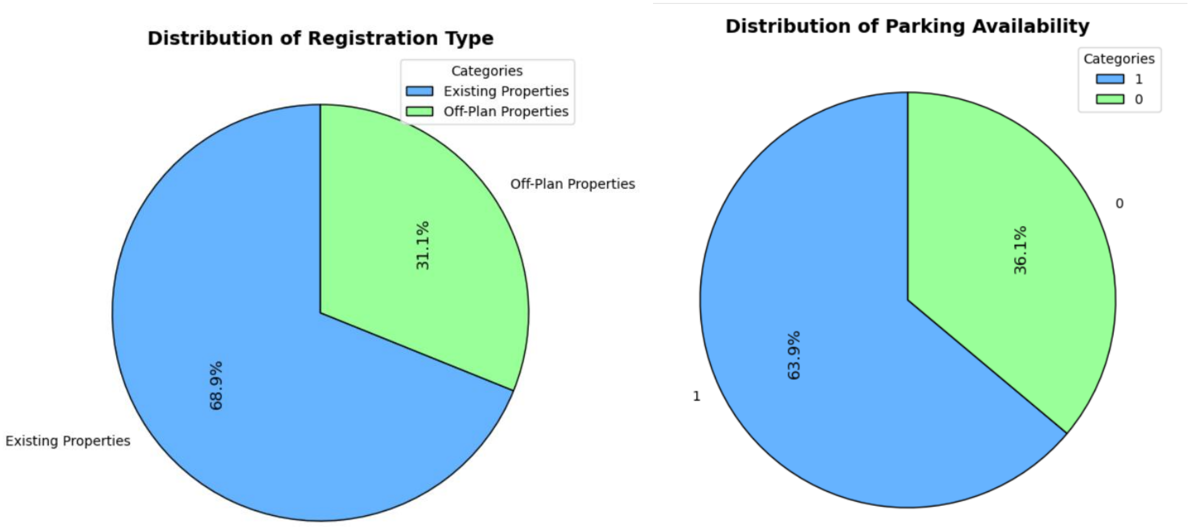


Fig 13. Distribution of Registration Type and parking availability

The figure above shows that 68.9% of registrations are for existing properties, while 31.1% are for off-plan properties. Additionally, 63.9% of properties have parking facilities, whereas the remaining 36.1% do not.

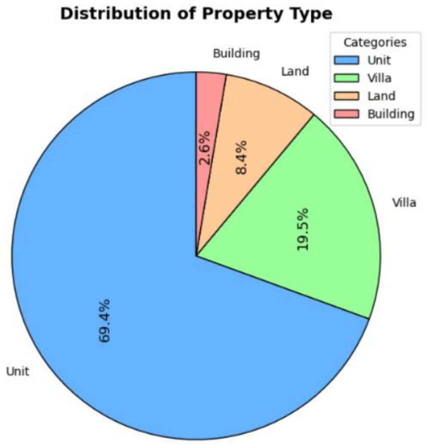


Fig 14. Distribution of Property Type

The dataset primarily consists of units (69.4%), emphasizing the dominance of apartments in the market. Villas (19.5%) hold a significant share, while land transactions (8.4%) represent a smaller segment. The remaining 2.6% include buildings or specialized properties like commercial spaces.

5.3 Bivariate Analysis

Bivariate analysis was conducted to explore relationships and dependencies between pairs of variables within the dataset.

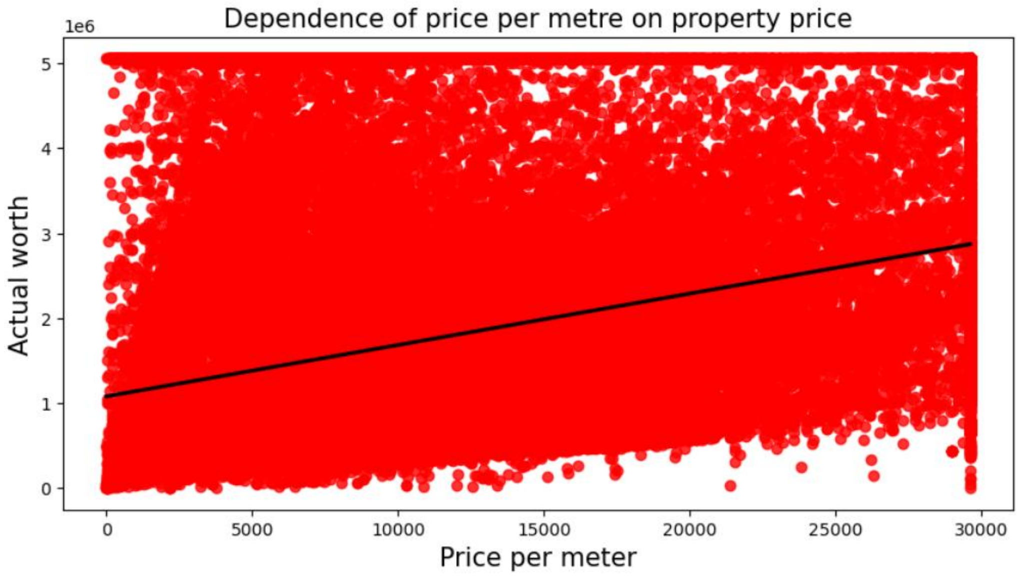


Fig 15. Relationship between Actual Worth and Meter Sale Price

The above figure shows significant variation in property values at different price points per square meter. The trend line indicates a positive correlation, suggesting that as the price per square meter rises, the total property worth generally increases, though with some fluctuation.

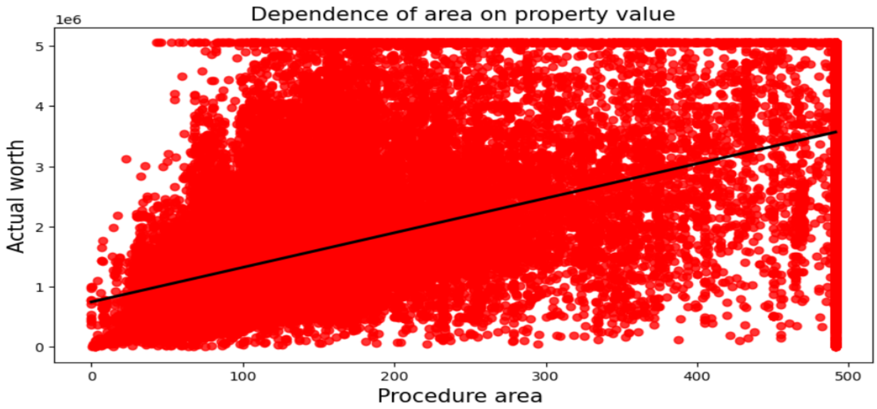


Fig 16. Relationship between Actual Worth vs Procedure Area

Figure 16 illustrates the relationship between Procedure Area and Actual Worth, showing a positive correlation where larger areas tend to have higher property values. However, the considerable variance observed indicates that area alone is not a reliable predictor, as other factors also play a significant role in determining property prices. Parking Availability is one of the key factors as shown below

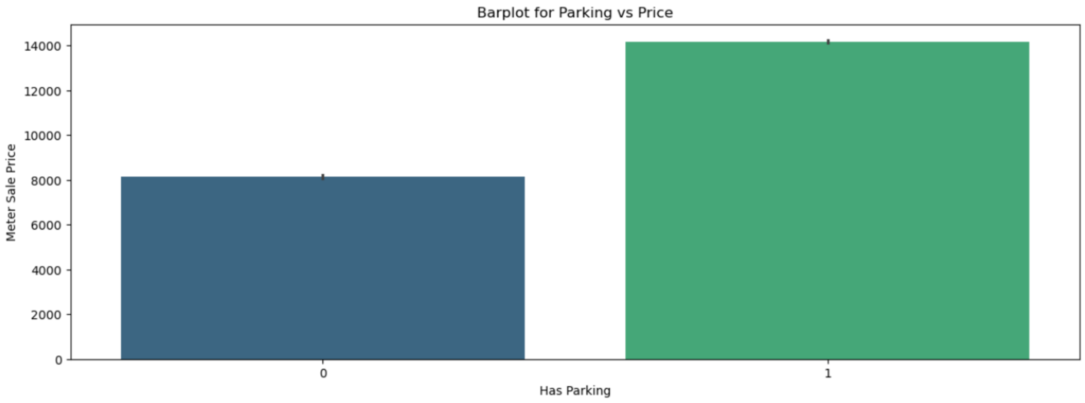


Fig 17. Relationship between parking availability and the meter price

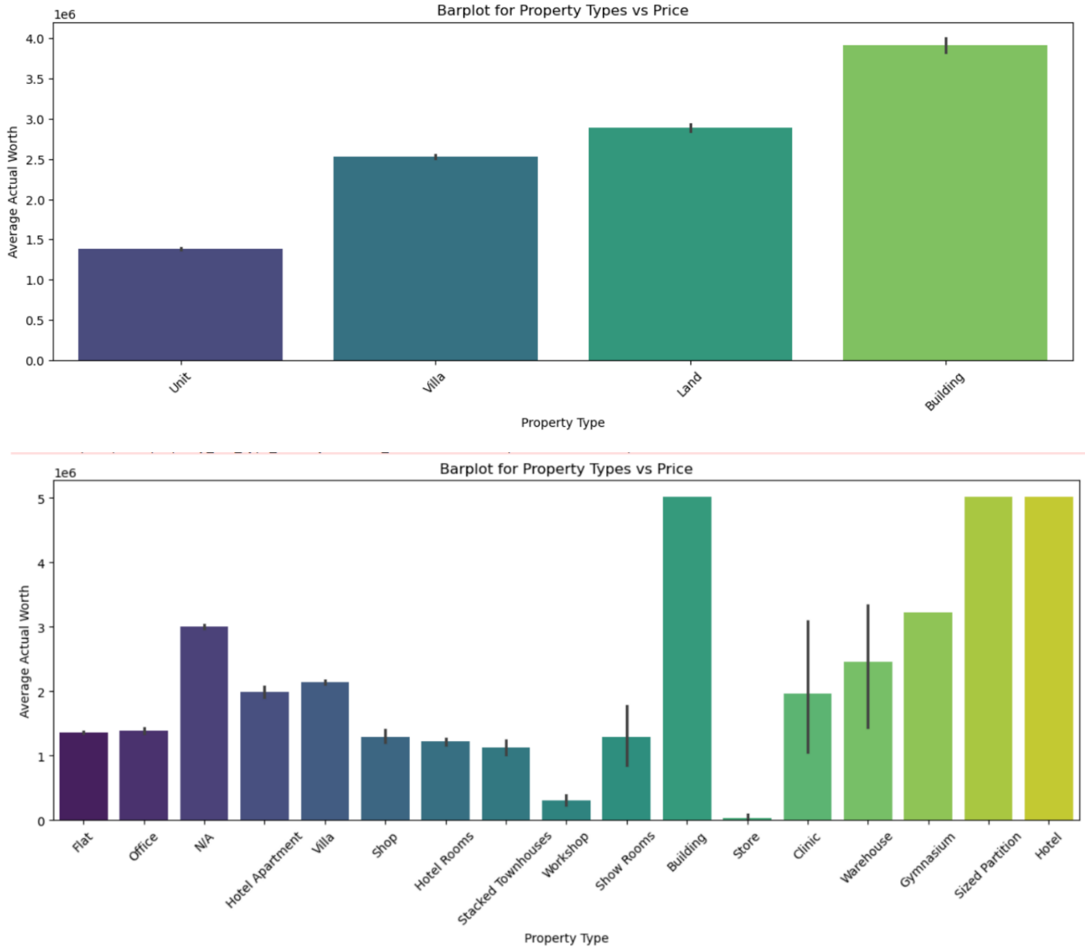


Fig 18. Relation between Actual Worth Vs Property Type and Property Subtype

The first plot (in the Fig 18.) reveals that Buildings have the highest average market value, followed by Land, Villas, and Units, indicating that larger property types like Buildings and Land generally command higher prices compared to smaller types like Units. Additionally, the second plot (Fig. 18), buildings and hotels tend to have the highest prices, while workshops and stores are priced the lowest. Flats, offices, and villas exhibit relatively stable pricing, whereas clinics and showrooms show greater price variability, suggesting that certain property types experience more market fluctuation than others.

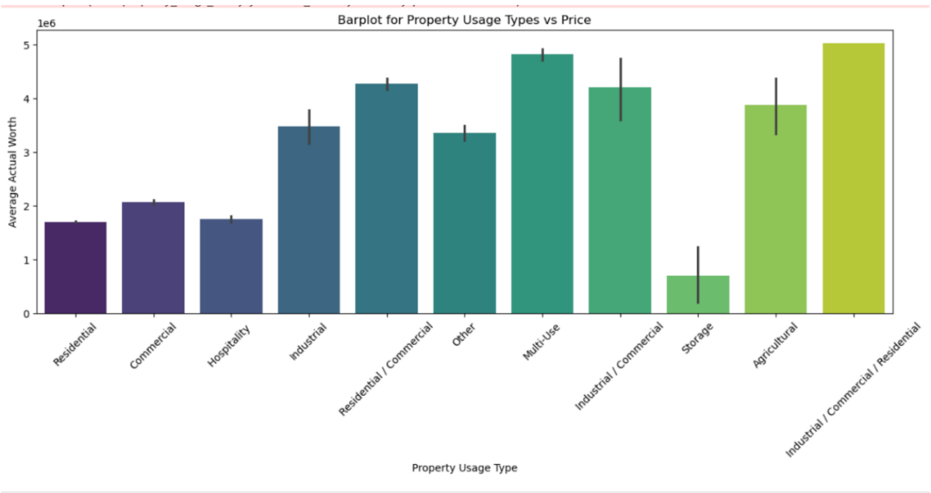


Fig 19. Relationship between Actual Worth and Property Usage

Figure 19 shows that mixed-use and industrial properties have the highest values, while storage properties hold the lowest prices. There is significant price variability in industrial and commercial properties, whereas residential and hospitality properties exhibit more stable pricing.

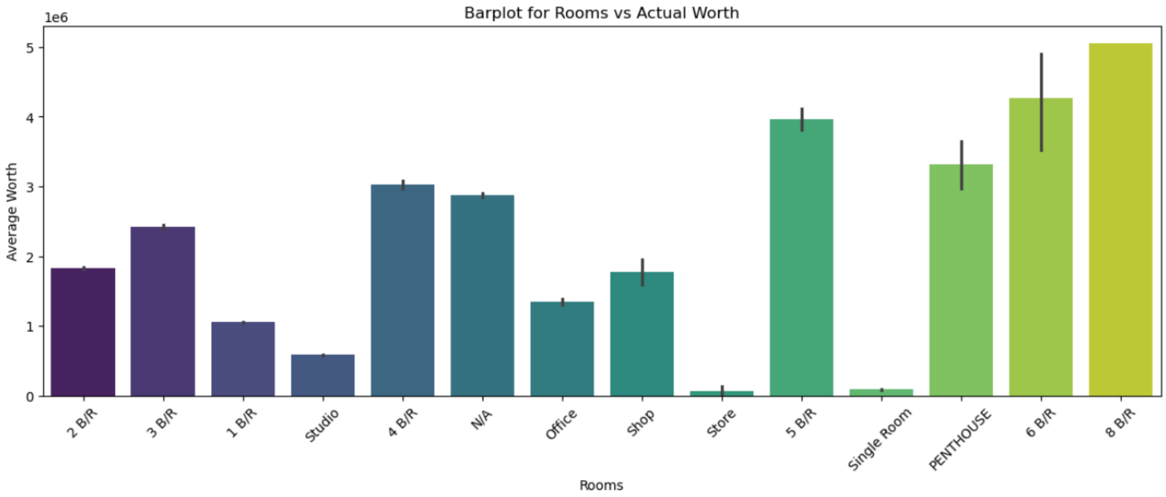


Fig 20. Relationship between Actual Worth and Number of Rooms

The trend shows that larger properties (with more bedrooms) tend to have higher actual worth, with 8-bedroom and penthouse units reaching the highest values. Meanwhile, studios and single-room properties exhibit the lowest worth. The presence of commercial spaces like offices and shops shows distinct pricing behavior compared to residential units.

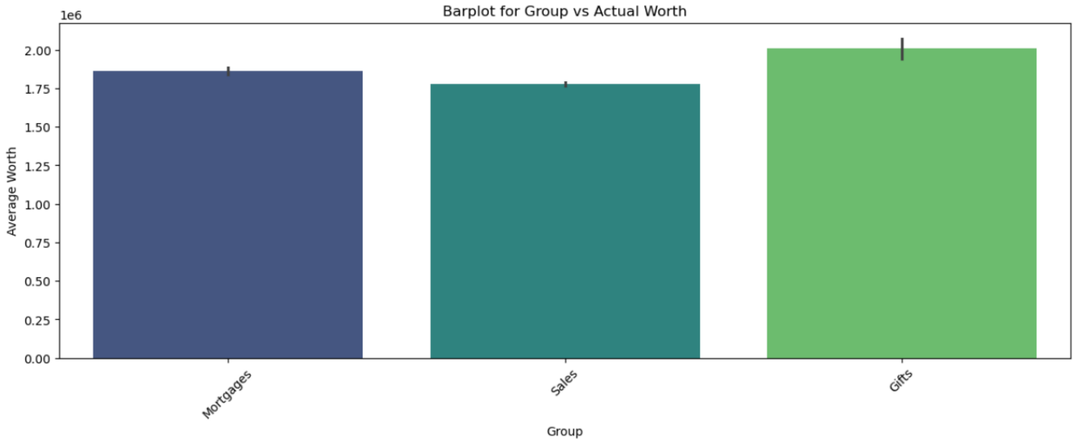
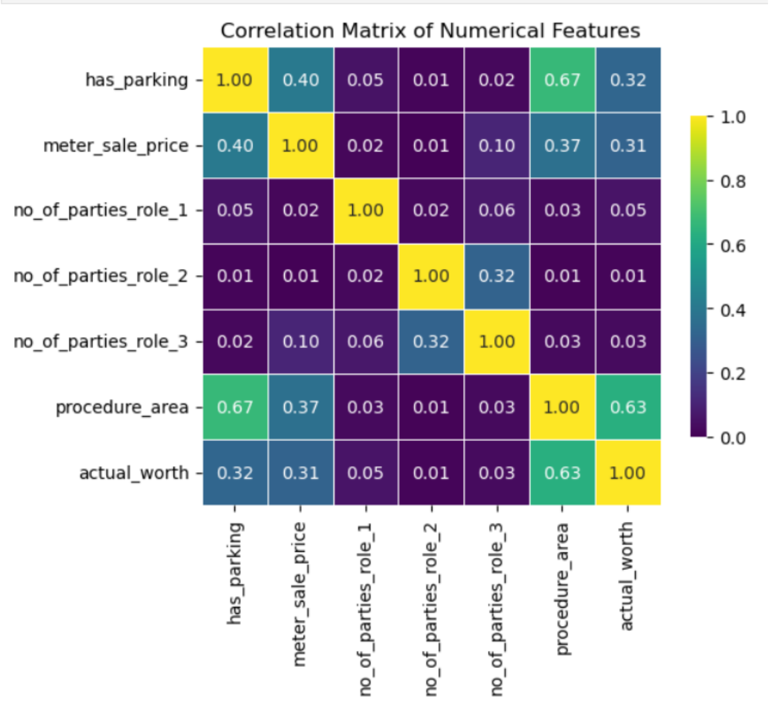


Fig 21. Relationship Between Actual Worth and Transaction Group

The "Gifts" category appears to have the highest average property worth, followed by mortgages and sales. This could suggest that gifted properties often belong to higher-value segments, while mortgage and sales transactions represent a more balanced distribution of property values.



	has_parking	meter_sale_price	no_of_parties_role_1	no_of_parties_role_2	no_of_parties_role_3	procedure_area	actual_worth
has_parking	1.00	0.40	0.05	0.01	0.02	0.67	0.32
meter_sale_price	0.40	1.00	0.02	0.01	0.10	0.37	0.31
no_of_parties_role_1	0.05	0.02	1.00	0.02	0.06	0.03	0.05
no_of_parties_role_2	0.01	0.01	0.02	1.00	0.32	0.01	0.01
no_of_parties_role_3	0.02	0.10	0.06	0.32	1.00	0.03	0.03
procedure_area	0.67	0.37	0.03	0.01	0.03	1.00	0.63
actual_worth	0.32	0.31	0.05	0.01	0.03	0.63	1.00

Fig 22. Correlation Matrix

Next, we explored the correlation matrix of numerical features using a heatmap (Fig. 22), which provides a visual representation of the strength and direction of relationships between the variables. The analysis revealed strong positive correlations between "procedure_area" and "has_parking" (0.67), as well as between "procedure_area" and "actual_worth" (0.63), suggesting that larger properties tend to have parking facilities and are generally valued higher. Additionally, "meter_sale_price" exhibited moderate correlations with both "procedure_area" (0.37) and "actual_worth" (0.31), indicating that while the sale price per meter is related to property size and worth, the relationship is not as strong. These insights are crucial for feature selection, as highly correlated variables may provide redundant information, and understanding these correlations helps assess potential multicollinearity issues, which can affect the performance of predictive models.

5.4 Models Performance

This section reviews the performance of each model using key evaluation metrics: R-squared, RMSE (Root Mean Squared Error), and MAE (Mean Absolute Error). These metrics provide a comprehensive understanding of model accuracy, with R-squared indicating the proportion of variance explained by the model, RMSE measuring the average magnitude of prediction errors, and MAE highlighting the average absolute difference between predicted and actual values. For deep learning models, we also examine the loss curve, which shows how the model's error decreases during training and helps assess convergence and the potential for overfitting. For machine learning models, we use scatter plots of actual vs. predicted values to visually compare predictions against true values, providing insights into the model's predictive accuracy. These plots are essential for identifying patterns or discrepancies in predictions and for evaluating the model's ability to generalize to unseen data. By examining these various visual and quantitative metrics, we can assess the strengths and limitations of each model in predicting real estate prices.

The results in Table 2 highlight the TabTransformer as the top-performing model, achieving the highest R^2 of 94.12%, RMSE of 344,958.34, and MAE of 159,078.81, demonstrating its superiority in handling tabular data. The Transformer model follows closely with an R^2 of 94.11%, RMSE of 345,240.38, and MAE of 153,204.39, showing a minimal 0.01% decrease in R^2, a slight 0.08% increase in RMSE, and a minor 0.03% decrease in MAE compared to the TabTransformer. The FT-Transformer performs solidly with an R^2 of 93.22%, but its higher RMSE (370,530.96) and MAE (192,833.38) indicate room for improvement, reflecting a

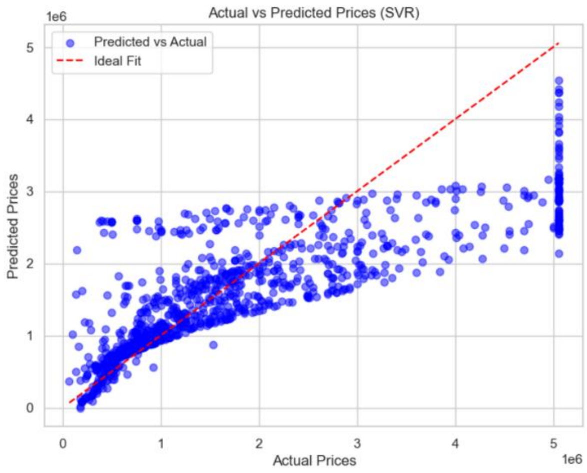
0.95% decrease in R², a 7.4% increase in RMSE, and a 21.3% increase in MAE compared to the TabTransformer.

Table 2. Evaluation Metrics of different Models

Models	R^2	MAE	RMSE
Transformer	**0.941135**	**153204.390948**	**345240.380448**
TabTransformer	**0.941231**	**159078.810794**	**344958.337687**
FT-Transformer	**0.932195**	**192833.382783**	**370530.961886**
GRU	0.927371	202045.598969	383484.412199
DCN	0.902581	247148.865981	444135.805982
SVR	0.608007	567974.863084	867442.661053
RF	0.899815	153405.969861	438533.852730
GBR	0.903424	164829.770647	430563.383624
XGBoost	0.900025	153571.368843	438073.089860
Ridge Regression	0.753162	466908.205486	688346.886731
Lasso Regression	0.753150	467099.192273	688364.592799

The traditional machine learning models show varied performance, with SVR having the lowest results (R^2: 60.80%, RMSE: 867,442.66, MAE: 567,974.86), trailing the top deep learning models significantly. Random Forest (RF) performs better with an R^2 of 89.98%, MAE of 153,405.97, and RMSE of 438,533.85, reflecting a 4.14% gap below the TabTransformer. Gradient Boosting Regressor (GBR) and XGBoost follow closely with R^2 values of 90.34% and 90.00%, respectively, showing similar performance to RF but still around 3.78%-4.12% behind the TabTransformer. Ridge Regression and Lasso Regression have lower performance (R^2: 75.32%), with much higher MAE and RMSE, clearly falling short of the Transformer-based models. While ensemble models like RF, GBR, and XGBoost outperform the linear models, they still lag the deep learning models, particularly those based on the Transformer architecture, in capturing complex data relationships.

In summary, while the traditional ML models provide decent performance, particularly the ensemble models like RF, GBR, and XGBoost, they still fall significantly short in comparison to the TabTransformer, which demonstrates superior ability to capture the intricacies of the data and deliver higher predictive accuracy. The deep learning models, especially those based on the Transformer architecture, dominate in handling complex datasets like those encountered in this real estate price prediction task.



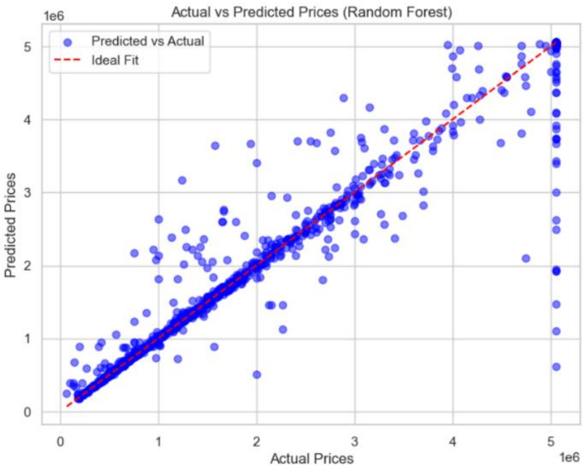
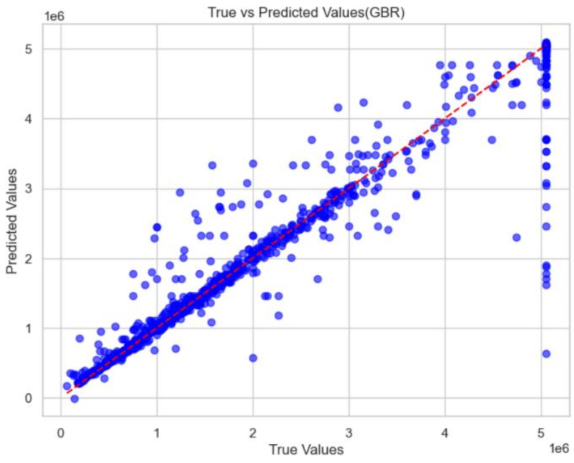


Fig 23: Scatter Plot for SVR

Fig 24: Scatter Plot for RF



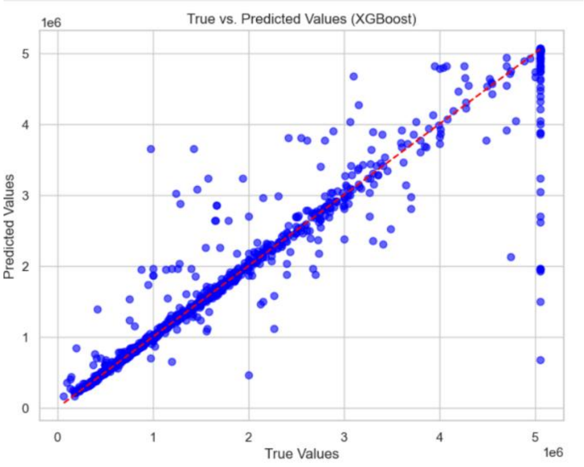
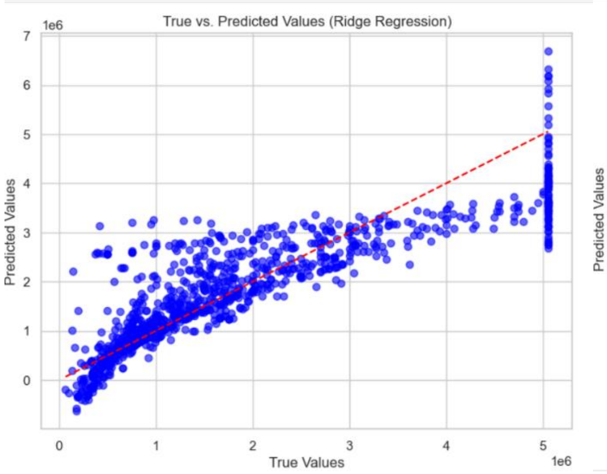


Fig 25. Scatter plot for GBR

Fig 26. Scatter plot for XGBoost



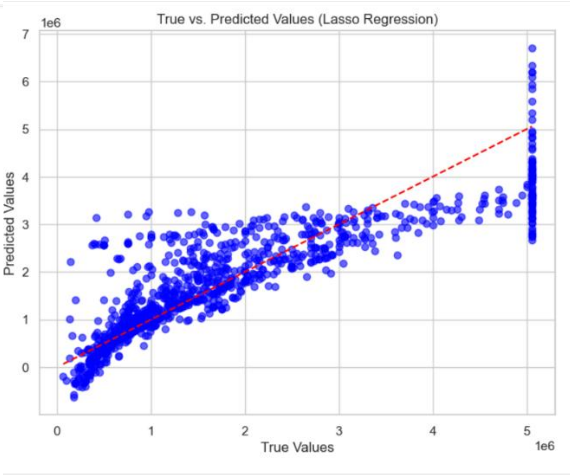
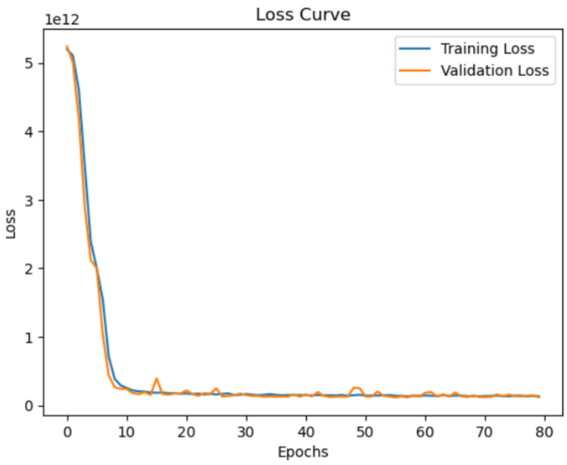


Fig 27. Scatter Plot for Ridge Regression Fig 28. Scatter plot for Lasso Regression

This scatter plot analysis visualizes, from Fig 23-28, the actual versus predicted property prices across multiple regression models, offering a clearer view of model performance across different price ranges compared to bar or line charts. By mapping individual data points, the scatter plot highlights areas where predictions align well with actual values and where models face challenges, particularly with high-value properties. Traditional linear models such as Support Vector Regression (SVR), Ridge, and Lasso exhibit difficulties in predicting high-value properties due to their inherent linear constraints, leading to increased variance and less accurate estimations in this segment. In contrast, ensemble methods like Random Forest improve prediction accuracy by capturing more complex patterns and generalizing better across different price ranges. However, minor deviations persist, particularly in high-end properties. Among the machine learning models evaluated, Gradient Boosting Regressor (GBR) and XGBoost demonstrate the best performance, effectively capturing non-linear relationships and reducing the impact of outliers. While slight deviations remain for higher-priced properties, these models consistently provide superior predictive accuracy. Further optimization through hyperparameter tuning, using advanced techniques like Bayesian Optimization (Optuna) or Hyperband (Ray Tune), can intelligently refine critical parameters. These methods enhance model efficiency and overall performance in real estate price prediction by optimizing hyperparameters more efficiently than traditional approaches.



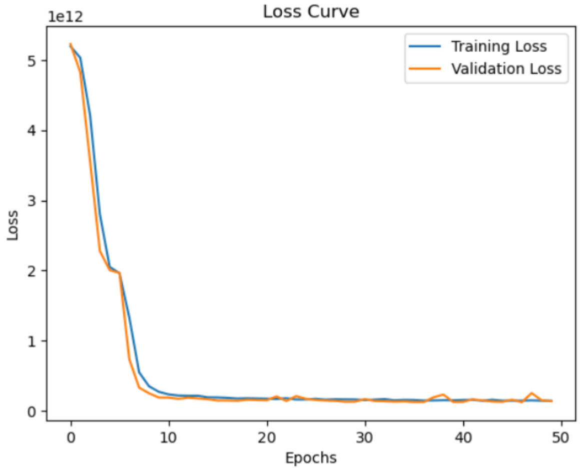
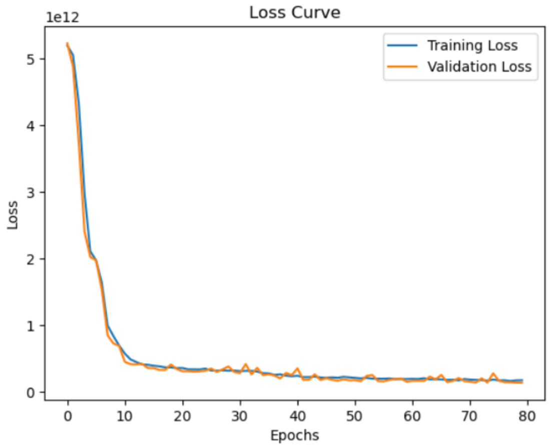


Fig 29. Loss Curve for Transformer

Fig 30. Loss Curve for TabTransformer



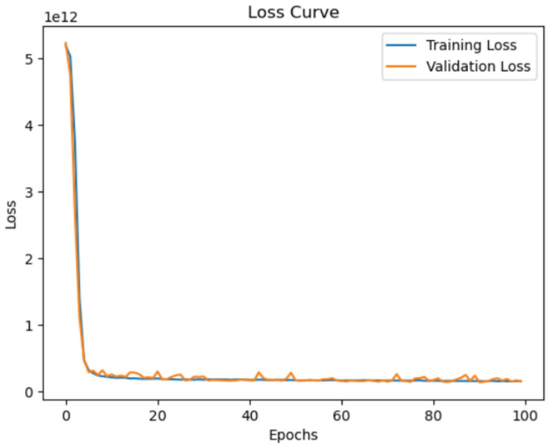


Fig 31. Loss Curve for FT-Transformer

Fig 32. Loss Curve for GRU

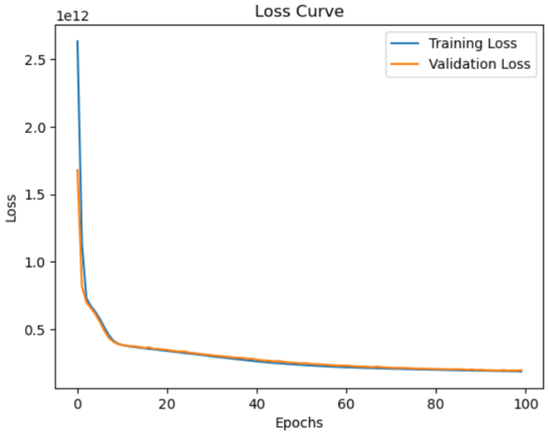


Fig 33. Loss Curve for DCN

The models—Transformer, TabTransformer, FT-Transformer, GRU, and DCN—each utilize distinct hyperparameter configurations to optimize performance in real estate price prediction.

The Transformer model leverages multi-head self-attention to efficiently capture complex dependencies between input features, making it highly effective in handling structured data with intricate relationships. TabTransformer extends this capability by enhancing categorical features learning through specialized embeddings, allowing for more effective representation of tabular datasets. FT-Transformer, an evolution of these models, further integrates positional encoding, improving adaptability for datasets containing both numerical and categorical features, ensuring more precise feature interactions. On the other hand, GRU (Gated Recurrent Unit), designed for sequential data processing, is particularly effective for time-series forecasting where historical trends influence predictions. Meanwhile, DCN (Deep & Cross Network) specializes in feature crossing, efficiently capturing non-linear interactions between variables, making it useful for structured datasets where feature combinations are critical. Among these models, TabTransformer achieved the highest accuracy in predicting Dubai real estate prices, outperforming other architectures due to its attention-based mechanisms and categorical feature representation, which allow for better learning of structured feature relationships.

The loss curves (Fig. 29–33) provide valuable insights into the training dynamics of these models, illustrating trends in convergence, overfitting, and underfitting. TabTransformer and Transformer demonstrated smooth convergence, indicating stable training and effective learning of patterns from the dataset. Unlike conventional evaluation metrics, loss curves offer direct insights into optimization challenges, helping to detect irregularities such as exploding gradients, poor generalization, or learning rate inefficiencies. In contrast, GRU and DCN exhibited occasional fluctuations, suggesting a need for further optimization. Techniques such as dropout regularization, adaptive learning rate tuning, and gradient clipping can be employed to improve their convergence and stability. To further refine TabTransformer's performance, advanced hyperparameter tuning strategies such as adjusting the number of attention heads, modifying batch sizes, and optimizing layer depths could enhance its generalization across diverse datasets. Additionally, leveraging state-of-the-art optimization techniques like Bayesian Optimization (Optuna) or Hyperband (Ray Tune) could significantly enhance model efficiency by automatically identifying the best hyperparameters, improving computational efficiency, and reducing training time. These refinements will help maximize predictive accuracy and ensure robust performance across different real estate market scenarios.

5.5 Challenges and Limitations

The challenges of using machine learning (ML) and deep learning (DL) models for predicting Dubai real estate prices include handling heterogeneous data, overfitting, and underfitting, especially with outliers and data variance. While DL models like TabTransformer capture complex relationships, they require large datasets and computational resources and may struggle with external factors like economic fluctuations and government policies. ML models, such as SVR and XGBoost, face difficulties with feature selection and data preprocessing, limiting their ability to model real estate price volatility. Both approaches also suffer from limited interpretability, making decision-making challenging, and may not effectively account for external variables, leading to uncertainty in predictions.

5.6 Summary

The results and analysis of Dubai real estate price prediction models show key insights from univariate and bivariate analyses, where larger areas typically lead to higher property values, and property types like buildings and land have higher market values. In terms of model performance, TabTransformer performed best with the highest R^2 (94.12%), RMSE (344,958.34), and MAE (159,078.81), followed closely by the Transformer model. While deep learning models like GRU and DCN struggled with tabular data, traditional ML models like SVR performed poorly, and ensemble models (RF, GBR, XGBoost) outperformed linear models but still lagged deep learning models. These findings underscore the effectiveness of attention-based models in capturing complex data relationships in real estate pricing.

CHAPTER 6

CONCLUSION AND RECOMMENDATION

6.1 Introduction

This section summarizes the findings from predicting Dubai real estate prices using ML and DL models, with a focus on the proposed model. The analysis assessed model performance based on features like numerical and categorical data. Attention-based models, such as the TabTransformer, outperformed traditional ML models in capturing complex relationships. Univariate and bivariate analyses reveal key correlations influencing prediction accuracy. While the proposed model, TabTransformer, showed strong performance, challenges remain due to data variability and external market factors. Future refinements could enhance the robustness and predictive accuracy of the proposed model, addressing these challenges for better real estate price predictions.

6.2 Conclusion and Recommendation

The evaluation of Dubai's real estate price prediction models underscores the superiority of deep learning architectures over traditional machine learning (ML) models, particularly in capturing complex market trends and relationships between features.

Among the ML models tested, Support Vector Regression (SVR) performed the worst (R^2 = 0.608, RMSE = 867,442), struggling to generalize well, especially for high-value properties. Ridge and Lasso Regression (R^2 = 0.753) showed moderate improvements but remained insufficient due to their linear nature, which limits their ability to capture the non-linearity inherent in real estate pricing. Random Forest (R^2 = 0.90) demonstrated better performance, effectively handling non-linearity and feature interactions, but still exhibited some overfitting tendencies.

Gradient Boosting Regression (GBR) and XGBoost (both R^2 = 0.90) emerged as the best-performing ML models, benefiting from ensemble learning and tree-based feature selection. Among these, GBR achieved the lowest RMSE (430,563), indicating more precise predictions. While RandomizedSearchCV was used for hyperparameter tuning, modern optimization techniques such as Bayesian Optimization (Optuna) or Hyperband (Ray Tune) can further enhance performance by systematically exploring hyperparameter space while minimizing computational cost.

Deep learning models demonstrated outstanding predictive accuracy, significantly surpassing traditional ML approaches by effectively capturing both numerical and categorical feature interactions. TabTransformer emerged as the best performer ($R^2 = 0.941$, RMSE = 344,958), utilizing advanced attention-based mechanisms to model intricate feature relationships. The Transformer model ($R^2 = 0.941$) closely followed, leveraging self-attention and feed-forward layers to extract complex dependencies within the data. Other deep learning models also showcased strong performance, with FT-Transformer ($R^2 = 0.932$) integrating positional encoding to enhance adaptability for mixed data types. GRU ($R^2 = 0.927$) proved effective for modeling time-sensitive price fluctuations, making it particularly useful for sequential dependencies in real estate trends. Meanwhile, DCN ($R^2 = 0.902$) capitalized on deep and cross-layer networks to learn intricate feature interactions, reinforcing its capability in capturing non-linear dependencies. These models collectively highlight the power of deep learning in real estate price prediction, offering enhanced accuracy and adaptability over traditional ML techniques.

To further refine predictions and enhance model adaptability, several advancements can be implemented. Advanced hyperparameter tuning techniques, such as Bayesian Optimization (Optuna), Hyperband (Ray Tune), and Population-Based Training (PBT), can dynamically optimize hyperparameters, improving model efficiency while reducing computational overhead. Incorporating real-time and external data, including macroeconomic indicators (e.g., inflation, GDP growth, interest rates), location-based features (e.g., proximity to metro stations, schools, and business hubs), and market sentiment analysis using NLP models like BERT, can further enhance forecasting accuracy. Optimizing deep learning models by refining GRU with temporal embeddings and attention mechanisms can improve its ability to capture time-dependent property price fluctuations, while expanding DCN with higher-order feature crossings and neural architecture search (NAS) can strengthen feature interactions. Additionally, self-supervised learning (SSL) can be leveraged to pretrain models on large-scale real estate datasets, enhancing feature extraction and generalization. Model ensembling and hybrid approaches, such as stacked ensembles combining GBR, XGBoost, and TabTransformer, can integrate the strengths of multiple models, while a Transformer-GRU hybrid can leverage self-attention alongside sequential modeling for improved long-term trend analysis. By integrating these advancements, predictive models can achieve greater accuracy, adaptability, and robustness, making them valuable tools for investors, real estate developers, and policymakers navigating Dubai’s dynamic property market.

REFERENCES

Adetunji, A.B., Akande, O.N., Ajala, F.A., Oyewo, O., Akande, Y.F. and Oluwadara, G., (2021) House Price Prediction using Random Forest Machine Learning Technique. In: *Procedia Computer Science*. Elsevier B.V., pp.806–813.

Alfalasi, A., (n.d.) *House Price Prediction Using Machine Learning Model*. [online] Available at: https://repository.rit.edu/theses.

Aljuboori, A. and Abdulrazzq, M.M.A., (n.d.) *International Journal of Computing and Digital Systems Enhancing Accuracy in Predicting Continuous Values through Regression*. [online] Available at: http://journals.uob.edu.bh.

Alshamsi, A., (n.d.) *Prediction of Dubai Apartment Prices Using Machine Learning*. [online] Available at: https://repository.rit.edu/theses.

Alstadsaeter, A., Collin, M., Planterose, B., Zucman, G. and Økland, A., (2024) *Foreign investment in the Dubai housing market, 2020-2024*. [online] Available at: https://dubailand.gov.ae/en/eservices/inquire-whether-a-person-owns-a-.

Anon (2017) *2017 IEEE SmartWorld : Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI) : 2017 conference proceedings : San Francisco Bay Area, California, USA, August 4-8, 2017*. IEEE.

Anon (2019) *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE.

Anon (n.d.) K. Radhakrishnan et al / Analyzing House Sales Prices byhyperparameters tuning Method Using Deep Learning (DL) Techniques Analyzing House Sales Prices byhyperparameters tuning Meth. [online] Available at: www.neuroquantology.com.

Arabasy, M., Hussein, M.F., Abu Osba, R. and Al Dweik, S., (2024) Smart housing: integrating machine learning in sustainable urban planning, interior design, and development. *Asian Journal of Civil Engineering*.

Arumugam, T., Arun, R., Anitha, R., Swerna, P.L., Aruna, R. and Kadiresan, V., (2023) Advancing and methodizing artificial intelligence (AI) and socially responsible efforts in real estate marketing. In: *Data-Driven Intelligent Business Sustainability*. IGI Global, pp.48–59.

Baldominos, A., Blanco, I., Moreno, A.J., Iturrarte, R., Bernárdez, Ó. and Afonso, C., (2018) Identifying real estate opportunities using machine learning. *Applied Sciences (Switzerland)*, 811.

Borodulin, A., Gladkov, A., Gantimurov, A., Kukartsev, V. and Evsyukov, D., (2024) Using machine learning algorithms to solve data classification problems using multiattribute dataset. In: *BIO Web of Conferences*. EDP Sciences.

Calainho, F.D., van de Minne, A.M. and Francke, M.K., (2022) A Machine Learning Approach to Price Indices: Applications in Commercial Real Estate. *Journal of Real Estate Finance and Economics*.

Chiu, S.-M., Chen, Y.-C. and Lee, C., (2022) Estate price prediction system based on temporal and spatial features and lightweight deep learning model. *Applied Intelligence*, [online] 52, pp.808–834. Available at: https://doi.org/10.1007/s10489-021-02472-6.

Despotovic, M., Koch, D., Stumpe, E., Brunauer, W.A. and Zeppelzauer, M., (2023) Leveraging supplementary modalities in automated real estate valuation using comparative judgments and deep learning. *Journal of European Real Estate Research*, 162, pp.200–219.

Elnaeem Balila, A. and Shabri, A. Bin, (2024) Comparative analysis of machine learning algorithms for predicting Dubai property prices. *Frontiers in Applied Mathematics and Statistics*, 10.

Ezzeddine, I., (2024) Enriching Real-Estate Through Sustainable Public Spaces: The case of Dubai- UAE. *Emirati Journal of Business, Economics, & Social Studies*, 31, pp.121–132.

Fu, Q., (2022) Real Estate Tax Base Assessment by Deep Learning Neural Network in the Context of the Digital Economy. *Computational Intelligence and Neuroscience*, 2022.

Habbab, F.Z. and Kampouridis, M., (2024) An in-depth investigation of five machine learning algorithms for optimizing mixed-asset portfolios including REITs. *Expert Systems with Applications*, 235.

Hansun, S., Suryadibrata, A. and Sandi, D.R., (2022) Deep Learning Approach in Predicting Property and Real Estate Indices. *International Journal of Advances in Soft Computing and its Applications*, 141, pp.60–71.

Jafary, P., Shojaei, D., Rajabifard, A. and Ngo, T., (2024) *BIM and real estate valuation: challenges, potentials and lessons for future directions. Engineering, Construction and Architectural Management*, .

Jin, B. and Xu, X., (2024) Pre-owned housing price index forecasts using Gaussian process regressions. *Journal of Modelling in Management*.

Kang, J., Lee, H.J., Jeong, S.H., Lee, H.S. and Oh, K.J., (2020) Developing a forecasting model for real estate auction prices using artificial intelligence. *Sustainability (Switzerland)*, 127.

Kucklick, J.-P., Müller, J., Beverungen, D. and Mueller, O., (2021) *Association for Information Systems Association for Information Systems QUANTIFYING THE IMPACT OF LOCATION DATA FOR REAL QUANTIFYING THE IMPACT OF LOCATION DATA FOR REAL ESTATE APPRAISAL-A GIS-BASED DEEP LEARNING APPROACH ESTATE APPRAISAL-A GIS-BASED DEEP LEARNING APPROACH Recommended Citation Recommended Citation 'QUANTIFYING THE IMPACT OF LOCATION DATA FOR REAL ESTATE APPRAISAL-A GIS-BASED DEEP LEARNING APPROACH'*. [online] Available at: https://aisel.aisnet.org/ecis2021_rip/23.

Lee, H., Han, H., Pettit, C., Gao, Q. and Shi, V., (2024) Machine learning approach to residential valuation: a convolutional neural network model for geographic variation. *Annals of Regional Science*, 722, pp.579–599.

Lenaers, I., Boudt, K. and De Moor, L., (2024) Predictability of Belgian residential real estate rents using tree-based ML models and IML techniques. *International Journal of Housing Markets and Analysis*, 171, pp.96–113.

Al Marzooqi, F.I. and Redouane, A., (2024) Predicting Real Estate Prices Using Machine Learning in Abu Dhabi. *Iraqi Journal of Science*, 653, pp.1689–1706.

Mora-Garcia, R.T., Cespedes-Lopez, M.F. and Perez-Sanchez, V.R., (2022) Housing Price Prediction Using Machine Learning Algorithms in COVID-19 Times. *Land*, 1111.

Mostofi, F., Toğan, V. and Başağa, H.B., (2022) Real-estate price prediction with deep neural network and principal component analysis. *Organization, Technology and Management in Construction*, 141, pp.2741–2759.

Naz, R., Jamil, B. and Ijaz, H., (2024) *Machine Learning, Deep Learning, and Hybrid Approaches in Real Estate Price Prediction: A Comprehensive Systematic Literature Review. Proceedings of the Pakistan Academy of Sciences: Part A*, .

Rey-Blanco, D., Arbués, P., López, F.A. and Páez, A., (2024) Using machine learning to identify spatial market segments. A reproducible study of major Spanish markets. *Environment and Planning B: Urban Analytics and City Science*, 511, pp.89–108.

Rimal, R., Rimal, B., Bhandari, H.N., Pokhrel, N.R. and Dahal, K.R., (2024) Real Estate Market Prediction Using Deep Learning Models. *Annals of Data Science*.

Shoukry Rashad, A. and Farghally, M.A., (n.d.) *The US Monetary Conditions and Dubai's Real Estate Market: Twist or Tango?*

Sultan, M., Issa, S., Dahy, B., Saleous, N. and Sami, M., (2024) Fifty years of land use and land cover mapping in the United Arab Emirates: a machine learning approach using Landsat satellite data. *Frontiers in Earth Science*, 12.

Suresh Yalgudkar, S. and V. Dharwadkar, N., (2022) A Literature Survey on Housing Price Prediction. *Journal of Computer Science & Computational Mathematics*, 123, pp.41–45.

Tekouabou, S.C.K., Gherghina, Ş.C., Kameni, E.D., Filali, Y. and Gartoumi, K.I., (2024) *AI-Based on Machine Learning Methods for Urban Real Estate Prediction: A Systematic Survey. Archives of Computational Methods in Engineering*, .

Walacik, M. and Chmielewska, A., (2024) Real Estate Industry Sustainable Solution (Environmental, Social, and Governance) Significance Assessment—AI-Powered Algorithm Implementation. *Sustainability (Switzerland)*, 163.

Walthert, L., Sigrist, F., Fahrländer, * and Raumentwicklung, P., (n.d.) *Deep learning for real estate price prediction We thank FPRE for providing the data and Manuel Lehner for the* *interesting discussions and suggestions*. [online] Available at: https://ssrn.com/abstract=3393434.

Wang, S., Chen, Y., Cui, Z., Lin, L. and Zong, Y., (n.d.) Diabetes Risk Analysis based on Machine Learning LASSO Regression Model. *www.centuryscipub.com*, [online] 4, p.2024. Available at: www.centuryscipub.com.

Wang, X., Qiao, Y., Xiong, J., Zhao, Z., Zhang, N., Feng, M. and Jiang, C., (2024) Advanced Network Intrusion Detection with TabTransformer. *Journal of Theory and Practice of Engineering Science*, 403, pp.191–198.

Wang, X., Takada, Y., Kado, Y. and Yamasaki, T., (2019) Predicting the attractiveness of real-estate images by pairwise comparison using deep learning. In: *Proceedings - 2019 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2019*. Institute of Electrical and Electronics Engineers Inc., pp.84–89.

Xu, X. and Zhang, Y., (2024) Office property price index forecasting using neural networks. *Journal of Financial Management of Property and Construction*, 291, pp.52–82.

Yazdani, M., (2021) Machine Learning, Deep Learning, and Hedonic Methods for Real Estate Price Prediction. [online] Available at: http://arxiv.org/abs/2110.07151.

Yoshida, T., Murakami, D. and Seya, H., (2022) Spatial Prediction of Apartment Rent using Regression-Based and Machine Learning-Based Approaches with a Large Dataset. *Journal of Real Estate Finance and Economics*.

Zhan, C., Wu, Z., Liu, Y., Xie, Z. and Chen, W., (2020) Housing prices prediction with deep learning: An application for the real estate market in Taiwan. In: *IEEE International Conference on Industrial Informatics (INDIN)*. Institute of Electrical and Electronics Engineers Inc., pp.719–724.

Zhang, X., Ma, Y. and Wang, M., (2024) An attention-based Logistic-CNN-BiLSTM hybrid neural network for credit risk prediction of listed real estate enterprises. *Expert Systems*, 412.

Zhao, Y., Chetty, G. and Tran, D., (2022) Deep Learning for Real Estate Trading. In: *Proceedings of IEEE Asia-Pacific Conference on Computer Science and Data Engineering, CSDE 2022*. Institute of Electrical and Electronics Engineers Inc.

APPENDIX A

RESEARCH PROPOSAL

Abstract

Dubai's dynamic transformation has positioned it as one of the fastest-growing cities in the world, known for its luxury lifestyle, iconic architecture, and a booming real estate sector. As the city continues to attract international investors and ultra-high-net-worth individuals to invest in real estate, predicting real estate prices accurately becomes crucial for investors, brokers, stakeholders, buyers and sellers to make decisions on investing real Dubai real estate. This study explores the application of various advanced machine learning and deep learning algorithms in predicting Dubai's real estate prices. We evaluating the efficiency of performance of Machine learning models such as Support Vector Regression (SVR), Random Forest(RF), Gradient Boosting Regression (GBR), Extreme Gradient Boosting (XGBoost), Ridge and Lasso, against advanced deep learning models such as Deep and Cross Networks (DCN), Transformer, TabTransformer, Feature-Token Transformer (FT-Transformer) and Gated Recurrent Units (GRU) using evaluation matrices RMSE, MAE, R-squared. By examining these models, we aim to determine their effectiveness in predicting real estate prices and compare them to identify the most promising techniques for enhancing predictive accuracy in this dynamic market. It helps investors, brokers, and stakeholders to make decisions in the real estate market.

1. Background

Dubai is popular for fast-paced lifestyle, luxury cars, modern architecture, stunning skyscrapers, Burj Khalifa etc. These monumental changes have put Dubai into the center stage of the world in terms of civilization and to be named to be as one of the fastest growing cities in the world. The evolution of Dubai into a safe-haven real estate industry has attracted huge amounts of international ultra-high individuals to consider investing in this rapidly growing and ever-changing market. Predicting real estate prices is a challenging task due to the nature of the market. Prices are influenced by different factors like location, number of rooms, developer name, parking availability, property transaction type like off-plan or ready to move. Nowadays we can see that Dubai real estate is booming. In 1984, the leadership of Dubai, with its unique mindset which was always ahead of time declared the tax exemption, which was welcoming people into the country to trade freely and let them reap the benefits to the maximum which indirectly enhanced the economy. Most researchers have done price prediction using traditional Machine learning models. Traditional forecasting models may fail in predicting prices due to their dependent on linear assumption and lack of ability to handle large and complex data. So, there is a need for advanced ML and DL models to handle complex datasets like real estate to find out the deep pattern and complex relationship between the features.

Price prediction using advanced ML and DL models gives more accurate results than traditional linear models in a way that handling non-linear data, better performance in predicting results, feature selection in which DL and ML models can select more important features in the model building. Some research proved that ML models give accurate predictions, and some research proved that DL models give accurate predictions. In this study focuses on both ML and DL models. Comparing ML and DL models' performance can give an idea about which algorithms outperformed well. Evaluating their performance by using evaluation matrices MAE, RMSE and R-squared, comparing their strengths and weaknesses to find out the most enhancing predictive model which helps to make decisions. Accuracy in predicting price helps investors, brokers, buyers and sellers to make them decide to invest or hold off their property.

As Dubai has become one of the largest investment hubs in the world, investors and buyers are looking for the trends going on in the real estate sector. The above-mentioned study Linear regression1 model outperformed well (Alfalasi, n.d.). In a study, the ML model SVM is performed efficiently to predict Dubai real estate price (Elnaeem Balila and Shabri, 2024). The real estate transaction data is evolving over time and becomes large and complex. This proposed study is going to take the efficiency of deep learning models to predict the price. In a way the deep learning model can handle the complexity inside the features and the efficiency in selecting important driving features to build the model to minimize the inaccurate prediction values.

2. Problem statement

Various researches have been done about Dubai real estate price prediction by using Machine Learning models. But research in price prediction using deep learning is limited. In this study, focusing on both ML and DL models. Comparing real estate price prediction globally using deep learning gives much idea about how deep learning techniques try to find deep patterns with complex datasets like real estate than Machine Learning. During covid-19 time, the house prices were comparatively less because of lack of demand and travel restrictions. But by the end of 2022, there will be a very noticeable rise in price due to removing travel restrictions.

The above study uses the Dubai house price data from Kaggle with 20 different columns to find trends and predict price using LSVM (Linear Support Vector Machine), Generalized Linear Regression and Neural Networks with findings of Generalized Linear 1 model is best performing model with correlation 71.1% (Alfalasi, n.d.). Similarly, the study of Dubai Apartment prices using different ML algorithms like SVR, Random Forest, Decision tree, Linear regression, K-nearest neighbor, boosted regression model on the Dubai real estate dataset from Kaggle of 38 features with the findings Random Forest performs well with R-squared value 73.12% (Alshamsi, n.d.). There is another study to predict the price of 1 BHK apartment by using different modern Models like SVM, EEMD-SD-SVM, Random Forest, KNN, GBM and ANN. With the help of these models, they could find the trend going on and the factors which strongly to drive the price predictions with the evaluation matrices that minimize the prediction errors. EESD-MD-SVM model has achieved this minimal prediction error with R2 value 0.541, MSE=0.090, RMSE=0.3010 and MAPE=3.3310 (Elnaeem Balila and Shabri, 2024). Using a variety of machine learning models, including ensemble learning algorithms based on boosting (Gradient Boosting Regressor, Extreme Gradient Boosting, and Light Gradient Boosting Machine) and bagging (random forest and extra-trees regressor), the study examined house price prediction in Alicante, Paris, and found that no algorithm outperformed the others with 94,024 records (Mora-Garcia et al., 2022). Another study of predicting house prices using Random Forest technique with Boston housing dataset with 14 features. This study has come up with the fact that the housing prices are correlated with different factors like location, city, number of rooms, how old the property is etc. And the finding is the Random Forest model with R-squared value 90% (Adetunji et al., 2021). In the study of comparison analysis of ML and DL models against Hedonic regression models with the aim of reducing the human involvement in the price prediction to get better accuracy predictions using models. The study used different ML, DL and Hedonic models (ANN, KNN, BFPM, RF, SVM) and found out that the RF model outperformed against all other models with the R-squared value 86%, RMSE 0.21 and MAEP 1.14% (Yazdani, 2021). The study of price prediction using DL to investigate the DL models can give more prediction accuracy. The basic idea of deep learning models is to find the deep pattern and relationships between the features in the complex dataset The study uses transaction data of apartments in Switzerland which uses evaluation metrics as MSE and found out that meta model has the highest predictive accuracy with MSE 0.0332 (Walthert et al., n.d.). The estate price prediction system using DL uses 27,988 transactions records in Taiwan. The study proposed two frameworks to obtain the price prediction for both spatial and temporal features which are BEEP which uses CNN-LSTM model and LEPP which uses shallow RNN (Chiu et al., 2022). The study of price prediction using DNN, PCA-DNN (Principal Component Analysis-Deep Neural Network) which compares the benchmark model as SRA. The study uses evaluation matrices such as MAE, MSE and MAPE and found out that PCA-DNN outperforms well with 98% prediction accuracy (Mostofi et al., 2022). Another study happened to predict the real estate indices of 6 stocks included in the sector Liquid45(LQ45) by using bi-LSTM model. The study used matrices RMSE and MAPE and among the stocks BSDE, CTRA, PTPP, PWON, SMRA and WIKA, BSDE has highest accuracy value R-Squared value 98.86% (Hansun et al., 2022). These days price predictions using unstructured data like images also make sense. This study uses 1000 images from real estate portals to predict the price with the help of convolutional neural network which extracts information from visual features. They found out that when the sampling size increases the value of adj. R2 also increases (Despotovic et al., 2023). People are always attracted to visual or images. There is another study which uses image processing by using proposed method pairwise comparison method with the interior images of properties from At Home Co., Ltd. The proposed method achieved higher accuracy than other methods which is 68.8% (Wang et al., 2019). There is another study which uses CNN models to predict property prices with the help of 71,809 entries in which both master data (size, number of rooms/bathrooms, area etc.) and image data. The study concludes that combining master data and image data minimizes the prediction error and maximizes the accuracy with the value of RMSE 43,469 and MAE 37,337 (Kucklick et al., 2021). Another study uses DRL (deep reinforcement learning) with GAF and LSTM models for real estate trading with the data of publicly available Australian real estate data, found out that the proposed model DRL with the combination of GAF and LSTM provide high accuracy (Zhao et al., 2022). Similarly, a study uses DL algorithms BPNN and CNN to predict housing price with the open transaction dataset provided by the ministry of Taiwan and found out that CNN (Convolution Neural Network) performs better than BPNN with higher R-squared value 94.5% and lower RMSE, MAE, MAPE and RMSLE (Zhan et al., 2020). The office price prediction study with NN model by using data from Chinese real estate index system with findings of simple NN models performed in a stable way 1.45% average RMSE value (Xu and Zhang, 2024). Another study uses AI based ML techniques to predict the price and they have used ML techniques because the dataset is not large enough (Tekouabou et al., 2024). The study of real estate auction prediction by using AI to develop forecasting models. GA, ANN and regression analysis model are used with the s real estate auction cases of apartments in Seoul and found that the GA model is outperformed with MAPE is 8.86 and RMSE is 0.006 (Kang et al., 2020).The real market price prediction (real estate index s&p500-60)using GRU,LSTM,CNN and the GRU model gives the best predict among them with R value 99% (Rimal et al., 2024)

From the above-mentioned studies, some studies prove that ML models perform well in predicting real estate prices and some of them prove that DL model perform well in predicting real estate prices. In this study, Dubai real estate price prediction, using advanced ML and DL algorithms like SVR, RF, GBR, XGBoost, Ridge and Lasso, Deep and Cross Networks (DCN), Transformer, TabTransformers, Feature-Token Transformer (FT-Transformer) and Gated Recurrent Units (GRU) provides the ability to handle more complex dataset like real estate, can adapt various type of data and can predict more accurately comparatively simple linear models.

3.Aim and Objective

The main aim of this research is to conduct comparative analysis study of real estate price prediction using advanced ML and DL models. Advanced ML and DL is popular for accurate predictive results. These models can handle tabular data and effectively identify and select the most significant features to build prediction models.

The following objectives can be formulated from the aims as mentioned above.

- To gather historical data from relevant resources.
- To preprocess the data by addressing missing values, outliers and applying normalization techniques. Handle missing data by either removing less significant values or inputting them with mean, median or mode values. Identify and manage outliers using box plot or scatter plot. For normalization, apply Z-score normalization (Standardization) to bring the features to a common scale.
- To evaluate different ML and advanced DL models to determine which performs best.
- To evaluate the performance of proposed approach based on model evaluation matrices like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R-Squared.

4.Significance of the study

Predicting real estate prices is crucial these days. This study, Dubai real estate price prediction helps international investors, buyers, sellers, brokers to make decisions on investing on real estate. It also reduces the price bubble which means a rise in the price of a property above the actual market value without any fundamental factors supporting the price increase. So, predicting the more accurate price is very important. To achieve this, different machine learning and deep learning algorithms are proposed. Machine learning algorithms like eXtream Gradiant Boost (XGBoost) which is an effective algorithm for structured or tabular data like real estate and deep learning algorithm like Deep and cross network (DCN) which is used to identify the deep patterns in the real estate dataset.

5.Scope of the study

The scope of real estate price prediction study is wide. Collecting historical data and preprocessing and applying machine learning and deep learning algorithms to find which model is outperforming well. A comparative analysis study encourages machine learning and deep learning models. Understanding the trends going on in the current market helps investment decisions to make about buying, selling or holding own property. The price of a property depends on many factors like the area where the property is located, number of rooms, developer name, type of project like off-plan or ready to move.

Dubai has become one of the investment hubs. Real estate plays an important role in investments. When it comes to investing in real estate, always looking into the market trends. This study is helpful for investors, brokers and stakeholders to give useful insights in the real estate market.

6.Research Methodology

The focus of this study is a comparison of Machine Learning and Deep Learning algorithms of price prediction Dubai real estate. It involves different steps. The core of the methodology involves selecting and training machine learning and deep learning models. Much research has been done using traditional machine learning algorithms. In this study, deep learning algorithms are proposed to analyze the price prediction accuracy.

6.1 Dataset

The dataset we are using in this study is taken from Dubai Land Department (DLD) which is a public platform. This dataset contains comprehensive details about all types of Real-Estate Transactions. The dataset contains 1272712 records and 46 columns like property type, area name, transaction date, actual worth, rent value, developer name, availability of parking etc. Altogether size of 744.23 MB. The dataset contains more than 10 years of Dubai property prices. In this dataset the actual worth (price) of the property act as dependent or target variable with other features.

6.2 Data preprocessing

The dataset contains more than 10 years of transactional details. 1272712 records must be processed before training and building the models. The records are subjected to following steps

- The data is transformed into addressing missing values, outlier and inconsistencies to prevent inaccurate model predictions.
- Then the data should be normalized or standardized of numerical features and encoding of categorical variables.
- Next, feature engineering is conducted to create meaningful input variables from raw data, which can enhance the predictive power of the models.

After completing preprocessing steps, the dataset is ready to train and build the models.

6.3 Framework

Collecting historical data is the first step. Transaction dataset provided by Dubai Land Department to be preprocessed by handling missing value by imputing mean, median or mode values or imputing values and outlier handling by plotting box plot or scatter plot. Then apply normalization techniques such as standardization and min-max scaling to make features in a common scale. Then apply feature engineering for static features.

Model selection for ML, such as SVR, GBR, XGboost, RF and ridge and lasso and these models efficiently predict the result for regression type of tasks. For DL Transformer, Deep and Cross Networks (DCN), TabTransformers, Feature-Token Transformer (FT-Transformer), and Gated Recurrent Units (GRU) to predict the deep pattern and relationship between the features. Train and validate the models with evaluation matrices like MAE, RMSE, R-squared.

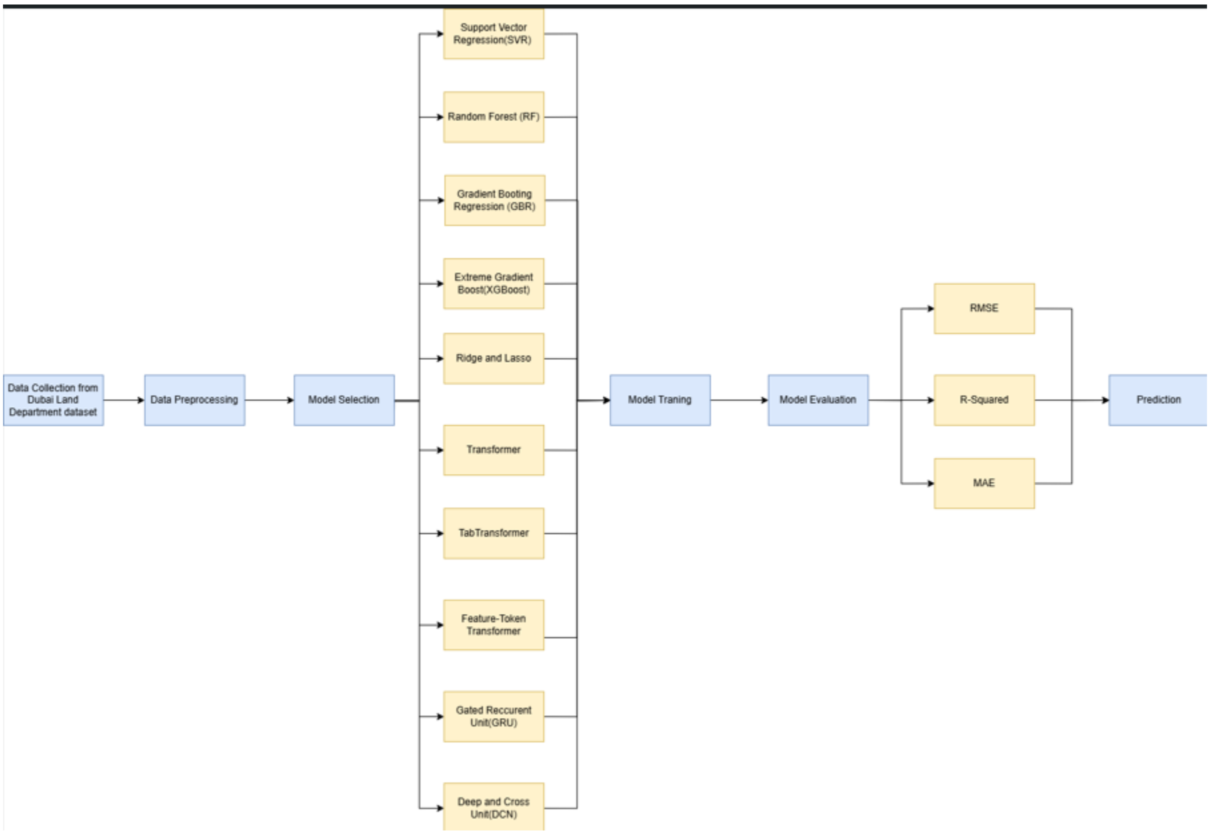

Data Collection from Dubai Land Department dataset
Data Preprocessing
Model Selection
Support Vector Regression(SVR)
Random Forest (RF)
Gradient Booting Regression (GBR)
Extreme Gradient Boost(XGBoost)
Ridge and Lasso
Transformer
TabTransformer
Feature-Token Transformer
Gated Reccurent Unit(GRU)
Deep and Cross Unit(DCN)
Model Traning
Model Evaluation
RMSE
R-Squared
MAE
Prediction

Fig.1 Propsed Framework

6.4 Support Vector Machine (SVM)

Support Vector Regression (SVR) is a powerful technique capable of handling both linear and non-linear regression tasks. The primary goal of the SVR algorithm is to find the optimal function that best fits the data within a certain margin of tolerance, rather than strictly classifying it. A key concept in SVR is the use of kernels, which help model complex relationships between the dependent and independent variables. Depending on the data pattern, an appropriate kernel—such as linear, polynomial, radial basis function (RBF), or sigmoid—must be selected. SVR effectively utilizes features such as property type, location, area, number of rooms and bathrooms, and property size to predict real estate prices. Once the relevant kernel is chosen, the model is constructed and trained using the available data.

6.5 Random Forest

A popular machine learning approach called random forest aggregates the output of several decision trees into a single outcome. It is an ensemble learning algorithm in which multiple decision trees have been made during the training the model and merges the outputs for making accurate predictions. Collecting samples followed by building models from these samples called bootstrap sampling. Individual decision trees are generated for each sample. Each decision tree produces output. Final output based on averaging for regression like predicting the price of the property from taking average price predictions from all trees. After training the model, RF can predict the price of property based on the features like size of the property, number of rooms, parking availability and cross validation can be applied to evaluate the performance of the model.

6.6 Gradient Boosting Regression (GBR)

Gradient Boosting Regression is widely used for its high prediction accuracy and efficiency, particularly with large and complex datasets. It is specifically applied when the target variable is continuous, such as property price. In Gradient Boosting Regression, the goal is to minimize a loss function using gradient descent by sequentially adding weak learners, typically decision trees. The process begins by initializing a base model, usually with the average of the target variable. Then, residuals—defined as the differences between actual and predicted prices—are computed. A new model is trained on these residuals to correct previous errors. The output for each decision tree leaf is calculated, and the existing model is updated accordingly. This process is repeated iteratively until no further performance improvements are observed. Gradient Boosting Regression also effectively identifies key features like property size, area, and number of rooms, optimizing the model by focusing on the most influential variables.

6.7 Extreme Gradient Boosting (XGBoost)

XGBoost Regression is a powerful and computationally efficient approach that builds upon Gradient Boosting techniques. Known for its speed and scalability, XGBoost handles large datasets effectively, offers insightful feature importance analysis, and manages missing values seamlessly. In regression tasks, such as predicting property prices, XGBoost begins by initializing the model with the mean of the target variable. It then calculates residuals—the differences between actual and predicted prices—and sequentially adds shallow decision trees to correct these errors. These shallow trees help control overfitting while still allowing the model to learn complex patterns. The algorithm minimizes the objective function, which includes both the loss function and a regularization term, by optimizing model predictions in each iteration. Through this iterative process, XGBoost continuously updates and improves the model until residual errors are minimized, ultimately training a robust regression model with an optimized loss function.

6.8 Ridge and Lasso

Ridge and Lasso model are one of the machine learning models. In ridge regression, an L2 penalty is added to reduce overfitting. In lasso regression adds L1 penalty to shrink coefficients and feature selection. Combine L1 and L2 penalties to balance the benefits of ridge and lasso. This model starts with building simple linear model to predict the target variable as price from the independent variables like size of the property, number of rooms, area etc. Then calculate the the loss function by comparing the actual price and predicted price. The Ridge models focus on price prediction and Lasso models focus on the feature selection like size of the property, location, number of rooms etc.

6.9 TabTranformer

TabTransformer is a deep learning model which is designed for handling tabular data. The idea behind TabTranformer is Transformer self-attention mechanism to handle deep complex relationship in the data. After preprocessing the data with handling missing values and outliers and normalization, the processed data passing through embedded layer to transform dense vector which is a crucial step. The transformer block, which is applies self-attention to embedded data, is allows the model to find the relationship between independent variable like size of the property, number of rooms, area, developer name etc. and target variable which is price of the property. Then these findings pass through the output layer to make predictions. Training the data with optimized loss function and optimizer. Then evaluate the model with evaluation matrices.

6.10 Feature-Token Transformer (FT-Transformer)

Feature-Token Transformer is a deep learning model which is extension of Transformer to handle tabular data. In FT-Transformer, each feature in dataset or tabular data converted into tokens. After data being normalized, the FT-Transformer utilizes its self-attention mechanism to understand the deep pattern and relationship between the independent variable like developer name, size of the property, number of rooms and target variable such as price. Then a stack of Transformer layers runs over the tokens. Finally, the output layer aggregates the refined feature for model prediction which is regression model for predicting the price of the property. Then training the data with optimized loss function and optimizer. Evaluate the model with evaluation matrices.

6.11 Transformer

The Transformer model, originally designed for sequence modelling, has been effectively adapted for tabular data prediction tasks. In this context, the Transformer handles tabular data by embedding categorical features and normalizing numerical ones, like other deep learning approaches. It leverages self-attention mechanisms to dynamically weigh the importance of different features for each data instance. This allows the model to focus on the most relevant attributes—such as property area, developer name, property type, and location—when making predictions. Unlike traditional models, the Transformer can capture complex interactions between features through multi-head attention layers. These layers help aggregate contextual information across all features, improving prediction accuracy. The model is trained using an optimized loss function and appropriate optimizer, and its performance is evaluated using standard evaluation metrics. The Transformer's ability to model global dependencies makes it a powerful tool for real estate price prediction using tabular data.

6.12 Deep and Cross Network (DCN)

Deep and Cross Network (DCN) is deep learning techniques which combine feature interactions to handle complex data. It contains two components, deep network and cross network. Categorical features like property type, parking availability are embedded into dense vectors and numerical features like the number of rooms and bathrooms and size are normalized. Then embedded features go through connected fully layers. Each cross layer takes output of the previous layer and applies linear transformation to find interaction between the features. The final layer of the connected layers produces the output as regression which is price of the property. Training the model with optimized loss function.

6.13 Gated Recurrent Units (GRU)

Gated Recurrent Unit (GRU) is an advanced RNN (Recurrent Neural Network) model. In GRU, there are three gates namely Reset gate, update gate and Forget gate. GRU takes two inputs, namely vector input and previous hidden state. Previous hidden state which combines current input and modified previous hidden state Finally we apply activation function. Activation function output ranges from 0 to 1 which will be caused by the gates to control the information flow. Finally training the model with optimized loss function. GRU is suitable for predicting trends and price patterns in Dubai real estate over time by analyzing historical transactions.

7.Required Resources

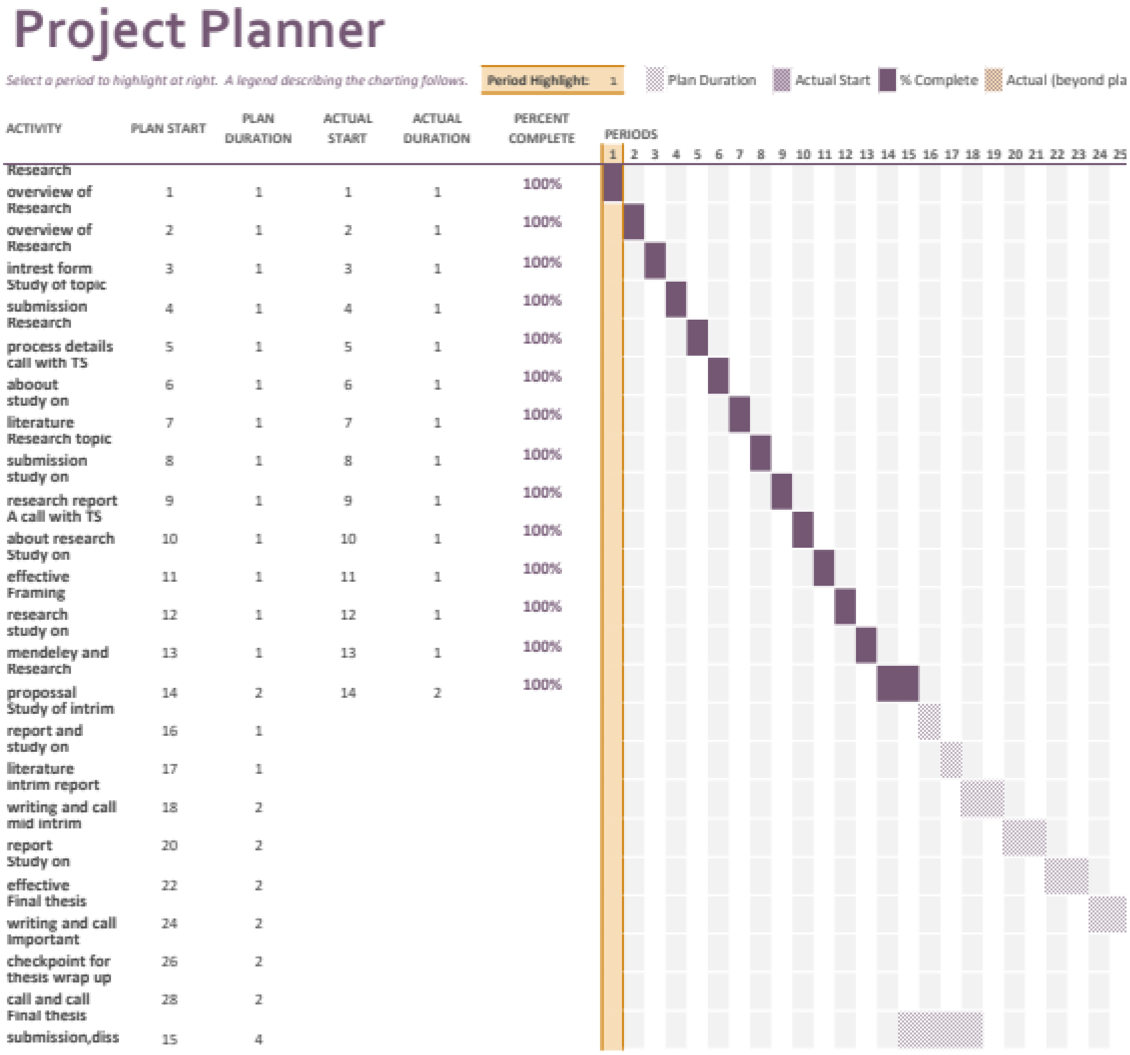
7.1 Software Requirements

- Python
- IDE (integrated Development Environment)-Jupyter notebook
- Required tools for building DL model- Pytorch
- Important libraries for preprocessing the data- Numpy, Pandas, Scikit learn, Matplotlib

7.2 Hardware requirements

- CPU
- Memory
- Storage
- Internet connectivity

8.Research Plan



Project Planner

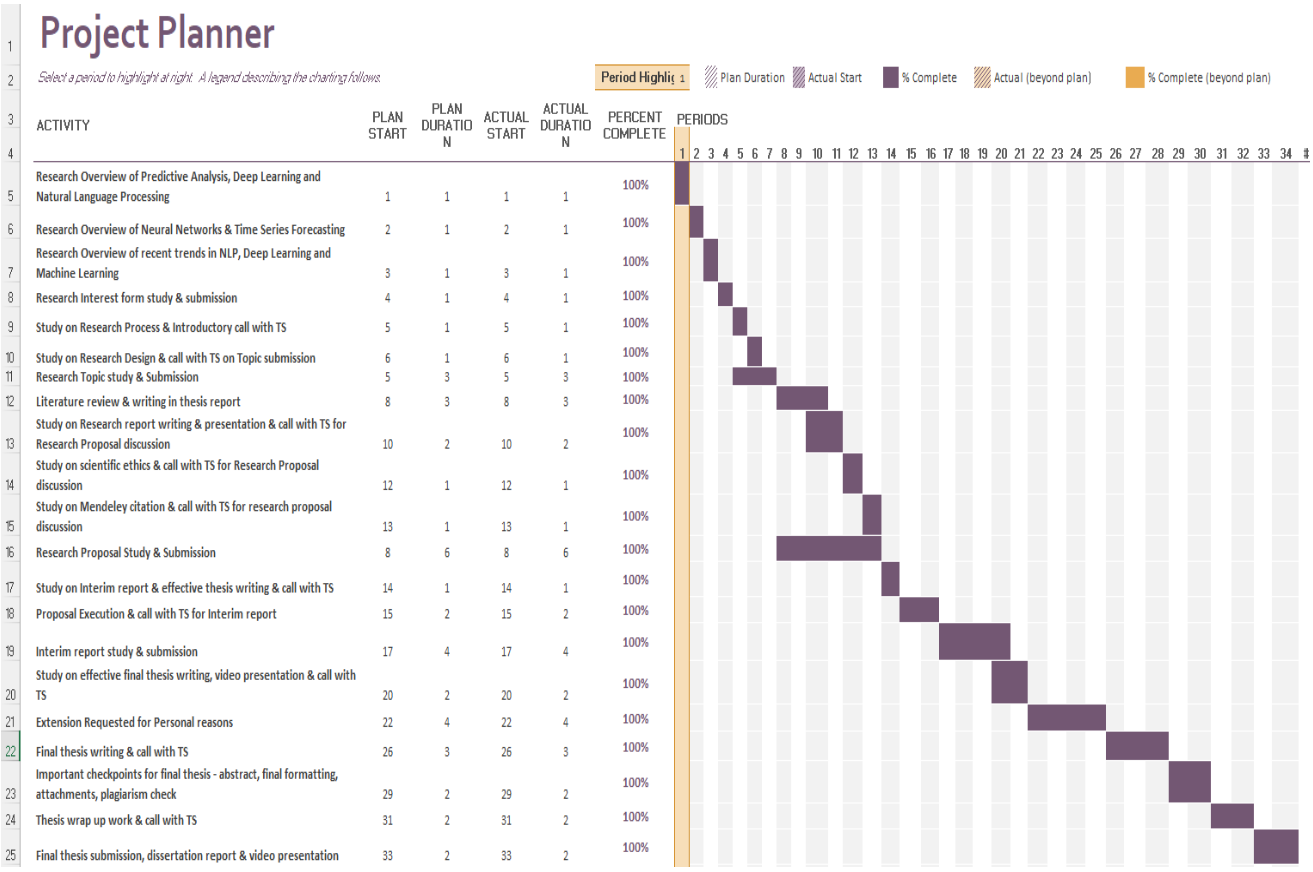
Select a period to highlight at right. A legend describing the charting follows.

Period Highlight: 1 ▨ Plan Duration ▨ Actual Start ■ % Complete ▨ Actual (beyond pla

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Research overview of	1	1	1	1	100%
Research overview of	2	1	2	1	100%
Research intrest form	3	1	3	1	100%
Study of topic submission	4	1	4	1	100%
Research process details	5	1	5	1	100%
call with TS aboout	6	1	6	1	100%
study on literature	7	1	7	1	100%
Research topic submission	8	1	8	1	100%
study on research report	9	1	9	1	100%
A call with TS about research	10	1	10	1	100%
Study on effective	11	1	11	1	100%
Framing research	12	1	12	1	100%
study on mendeley and	13	1	13	1	100%
Research propossal	14	2	14	2	100%
Study of intrim report and	16	1			
study on literature	17	1			
intrim report writing and call	18	2			
mid intrim report	20	2			
Study on effective	22	2			
Final thesis writing and call	24	2			
Important checkpoint for	26	2			
thesis wrap up call and call	28	2			
Final thesis submission,diss	15	4			

APPENDIX B

RESEARCH PLAN



Project Planner

Select a period to highlight at right. A legend describing the charting follows.

Period Highlig 1 ▨ Plan Duration ▨ Actual Start ■ % Complete ▨ Actual (beyond plan) ■ % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Research Overview of Predictive Analysis, Deep Learning and Natural Language Processing	1	1	1	1	100%
Research Overview of Neural Networks & Time Series Forecasting	2	1	2	1	100%
Research Overview of recent trends in NLP, Deep Learning and Machine Learning	3	1	3	1	100%
Research Interest form study & submission	4	1	4	1	100%
Study on Research Process & Introductory call with TS	5	1	5	1	100%
Study on Research Design & call with TS on Topic submission	6	1	6	1	100%
Research Topic study & Submission	5	3	5	3	100%
Literature review & writing in thesis report	8	3	8	3	100%
Study on Research report writing & presentation & call with TS for Research Proposal discussion	10	2	10	2	100%
Study on scientific ethics & call with TS for Research Proposal discussion	12	1	12	1	100%
Study on Mendeley citation & call with TS for research proposal discussion	13	1	13	1	100%
Research Proposal Study & Submission	8	6	8	6	100%
Study on Interim report & effective thesis writing & call with TS	14	1	14	1	100%
Proposal Execution & call with TS for Interim report	15	2	15	2	100%
Interim report study & submission	17	4	17	4	100%
Study on effective final thesis writing, video presentation & call with TS	20	2	20	2	100%
Extension Requested for Personal reasons	22	4	22	4	100%
Final thesis writing & call with TS	26	3	26	3	100%
Important checkpoints for final thesis - abstract, final formatting, attachments, plagiarism check	29	2	29	2	100%
Thesis wrap up work & call with TS	31	2	31	2	100%
Final thesis submission, dissertation report & video presentation	33	2	33	2	100%