Developing an AI-Powered Platform for Rental and Real Estate Search

Zainab Al Jammal

A Thesis Submitted to the Lebanese University, Faculty of Sciences
In Partial Fulfillment of the Requirements
for the Degree of Master of Science

Professional Program in Artificial Intelligence

Supervisor:
Dr. Hasan Harb

Lebanese University
Beirut, Lebanon
September 22, 2025

Abstract

Real estate relies heavily on data analysis, yet the quality of the data often hinders it. This project confronts the dual challenges of developing predictive models, and performing data-driven real estate analysis from a severely fragmented web-scraped property dataset and a small, high-noise historical transaction dataset and property listings characterized by extreme regional sparsity.

A pragmatic, two-phase approach was adopted. First, for price estimation, initial ensemble models failed ($R^2 = 0.2$) due to extreme regional sparsity and class imbalance. The scope was strategically narrowed to the only viable segment: apartments, where a robust XGBoost model was developed, achieving a viable R^2 of 0.78 and a Mean Absolute Error (MAE) of $88,475. For transaction forecasting, an initial LSTM model was rejected due to poor performance and overfitting, confirming its unsuitability for the limited data, leading to a fine-tuned, robust XGBoost model, incorporating seasonal features and strong regularization, it successfully captured broad market rhythms, resulting in a forecast with a Mean Absolute Percentage Error (MAPE) of 22% on the test set.

Second, to overcome the limitations of the sparse time-series data, a pivot was made from prediction to simulation. Five region-specific TimeGAN models were trained to generate realistic synthetic data, a method validated through interleaved PCA and t-SNE visualizations and strong quantitative metrics (Discriminative Score: 0.21, Predictive Score: 0.086).

The key outcome is a comprehensive web application thus integrating these facets in the final suite of analysis tools: A price estimator, a trend visualization, a chat assistant, a comparative market analysis (CMA) dashboard, a transaction forecast and, finally, a novel scenario simulator. This simulator leverages the synthetic data as a volatility-preserving baseline, applying dynamic, time-variant, and region-specific growth rates to model plausible "Boom" and "Crash" market trajectories, particularly in the context of Lebanon's post-2019 economic crisis.

This work demonstrates that in data-scarce environments, a dual strategy, combining direct AI application for prediction with synthetic data simulation for exploration, offers a resilient and practical framework for delivering advanced real estate intelligence in emerging markets.

Acknowledgment

I would like to express my deepest gratitude to all those who supported me throughout the journey of completing this thesis.

First and foremost, I sincerely thank my supervisor Dr. Hassan Harb for his invaluable guidance, honest feedback, and constant encouragement. His expertise, patience, and insightful suggestions not only shaped this research but also enriched my academic and personal growth as a researcher. I truly consider myself fortunate to have benefited from his mentorship and support.

I am also grateful to the faculty members of the Lebanese University, Faculty of Science, for providing me with the knowledge and academic foundation that made this work possible.

Special appreciation to my family. My parents' unwavering support has been my foundation, and without their encouragement and sacrifices, this achievement would not have been possible.

Finally, I extend my respect to my colleagues and friends who offered helpful discussions, shared resources, and provided moral support throughout my studies.

Table of Contents

ABSTRACT **0**

ACKNOWLEDGMENT **2**

CHAPTER 1: INTRODUCTION **8**

1.1) Background **8**

1.2) Problem and motivation **8**
1.2.1. The Time-Series Challenge: Forecasting on a Knife's Edge 9
1.2.2. The Tabular Data Challenge: Navigating a Landscape of Data Deserts 9

1.3) Research Questions and Objectives **10**

1.4) Project Overview and Thesis Statement **11**

2.1) AI Applications in Real Estate: Prediction and Valuation **12**
2.1.1. from traditional to modern valuation 12
2.1.2. Price Estimation Models 13
2.1.3. Transaction and Market Forecasting 15
2.1.4. AI Chat Assistant (Conversational Search) 16
2.1.5. Trend Visualization (City Circles + Price Trends) 17

2.2) The Pervasive Challenge: Data Scarcity and Quality in Real Estate Analytics **17**
2.2.1. The "Data-Rich, Information-Poor" Problem 17
2.2.2. Common Data Issues: Sparsity, Noise, Imbalance, and Historical Depth 18
2.2.3. The Privacy Conundrum 20

2.3) Synthetic Data Generation as a Strategic Solution **21**
2.3.1. Introduction to Generative Adversarial Networks (GANs) 21
2.3.2. GANs for Tabular and Time-Series Data 22
2.3.3. TimeGAN: The State of the Art for Sequential Data 23

2.4) Web Technologies for Data-Intensive Applications **24**

CHAPTER 3: METHODOLOGY **27**

3.1) System Architecture and Technology Stack **27**
3.1.1. Technology Stack Breakdown 28

3.2) Data Collection and Schema **29**
3.2.1. Web Scraping Process 29
3.2.2. Database Schema 30
3.2.3. A Peak in Database 31

3.3) Data preparation, Feature Engineering and Model Implementation **32**
3.3.1. General Cleaning 32

3.4) Price Estimation Model **35**
1. Dataset Context and Scope 35
2. Feasibility of Modeling Different Property Types 35
3. Data Cleaning and Preprocessing 38
4. Feature Engineering 38

3.5) Forecasting Transaction Model **41**
1. Dataset Context and Scope 41
2. Data Cleaning and Preprocessing 43
3. Feature Engineering 43
4. Noise and Outlier Strategy 44
5. Model Choice and Implementation 44

3.6) Chat Assistant **44**
1. Design and Logic 45
2. Functionality 45
3. Methodological Role 45

3.7) Trend Visualization (City Circles + Price Trends) **45**
1. Design and Components 45
2. Functional Role 46

3.8) Comparative Market Analysis (CMA) Tool **46**
1. Rational and Role 46
2. Implementation Logic 47
3. Computed Metrics 47
4. Workflow Summary 48
5. Significance 48

3.9) TimeGAN: A Framework for Generating Realistic Time-series Data **49**
1. Motivation and Core Challenge 49
2. Rational for Selecting TimeGAN 49
3. TimeGAN Architecture 50
4. The Joint Training Process 51
5. Application in This Research: Input Pipeline & Strategic Pivots 53

3.10) Scenario Simulator Design **54**
1. Privacy and Transparency 55
2. Theoretical Foundation: Scenario-Based "What-If" Analysis 56
3. Core Logic 56
4. Evolutionary Development: From Simple Curves to Narrative-Driven Simulation 57

CHAPTER 4: RESULTS AND FINDINGS **59**

4.1) Performance of Predictive Models **59**
4.1.1. Price Estimation Model Results 59
4.1.2. Transaction Forecasting Model Results 62

4.2) Evaluation Of The Synthetic Data Solution **65**
4.2.1. Inadequacy of Real Data For Simulation 65
4.2.2. TimeGAN Output Analysis 67

4.3) Scenario Simulator Outputs **71**

4.4) Web Application Showcase **72**
4.4.1. Fair Price Estimation 72
4.4.2. Transaction Forecasting 72
4.4.3. Chat Assistant Tool 73
4.4.4. Trend Visualization Results 77
4.4.5. Comparative Market Analysis (CMA) Tool 77
4.4.6. Market Scenario Simulator 78

CHAPTER 5: DISCUSSION AND ANALYSIS **79**

5.1) Apartment Price Estimation with XGBoost **79**

5.2) Transaction Forecasting under Data Scarcity **80**

5.3) Chat Assistant: Democratizing Access to Analytics **81**

5.4) Trend Visualization: From Data to Strategic Insight **81**

5.5) Synthetic Data Generation with TimeGAN **81**

5.6) Market Scenario Simulator: From Prediction to Simulation **83**

5.7) Synergy of AI and Traditional Tools (CMA as a Baseline) **84**

5.8) Limitations of the Study **85**

5.9) Broader Implications and Future Work **87**
5.9.1. Broader Implications and Practical Contributions 87
5.9.2. Future Work 87

CHAPTER 6: CONCLUSION **89**

6.1) Summary of Contributions **89**

6.2) Future Work **90**

REFERENCES **91**

Table of Figures

Figure 1: Regional Distribution of Apartment Listings 19
Figure 2: Generalized Architecture of a Generative Adversarial Network (GAN)............ 22
Figure 3: high-level overview of the modeling pipeline............ 25
Figure 4: System Architecture Diagram 28
Figure 5: Entity-relationship diagram............ 30
Figure 6: A high-level overview of the universal data cleaning pipeline 34
Figure 7: Distribution of All Property Types Across Districts 37
Figure 8: Stacked bar chart illustrating the distribution of property types across Lebanese provinces 37
Figure 9: Correlation heatmap of raw apartment dataset 38
Figure 10: Correlation heatmap of engineered apartment dataset 39
Figure 11: Raw monthly transaction values for Beirut (2011-Jan – 2016-Dec)............ 42
Figure 12: Smoothed monthly cross-city average transaction values 42
Figure 13: CMA workflow diagram 48
Figure 14: The TimeGAN Architecture............ 50
Figure 15: Top 20 feature importance by LightGBM model 60
Figure 16: Top 20 feature importance by XGBoost model............ 60
Figure 17: Regression plot by LightGBM model 61
Figure 18: Regression plot by XGBoost model............ 61
Figure 19: Top 20 feature importance by final XGBoost model 62
Figure 20: Regression plot by the final XGBoost model I 62
Figure 21: Model Validation for Beirut: Actual vs. Predicted by initial XGBoost model (2016 test year) 63
Figure 22: Model Validation for Beirut: Actual vs. Predicted by final updated XGBoost model (2016 test year)............ 64
Figure 23: Top feature importance by the final updated XGBoost model 64
Figure 24: Raw monthly transaction data for Baabda (2011-2016)............ 65
Figure 25: Raw monthly transaction data for Kesrouan (2011-2016)............ 66
Figure 26: Raw monthly transaction data for Tripoli (2011-2016) 66
Figure 27: Raw monthly transaction data for Bekaa (2011-2016)............ 67
Figure 28: t-SNE projection of real vs. synthetic data (failed single-model TimeGAN, merged cities) 68
Figure 29: Beirut Transaction Value: Original Data vs. Multi-City TimeGAN Synthes............ 68
Figure 30: t-SNE projection of real vs. synthetic data (successful per-city TimeGAN, e.g., Beirut)........... 69
Figure 31: Beirut Transaction Value: Original Data (2011-2016) vs. Single-City TimeGAN Synthetic Data (2017-2021)............ 70
Figure 32: 'Real Estate' web application's Market Scenario Simulator for Beirut (2017-2025) 71
Figure 33: Apartment Price Estimator............ 72
Figure 34: Transaction Forecasting page 73
Figure 35: Comparing Beirut and Jbeil Apartment Markets 74
Figure 36: Expected Transactions in Tripoli next year............ 75
Figure 37: Predicting price of apartments in Beirut in 2027............ 76
Figure 38: City Price Circles and Trends 77
Figure 39: The Comparative Market Analysis (CMA) dashboard............ 78
Figure 40: Market Scenario Simulator............ 78

Table of tables

Table 1: Distribution of property listings by type and region 10
Table 2: Comparative Overview of Real Estate Platforms 25
Table 3: Properties Table Schema 31
Table 4: Transaction Table Schema 31
Table 5: Snapshot of types of real estates and their corresponding count in the dataset Properties 35
Table 6: Annual average transaction value by city (2011–2016) 41
Table 7: Example CMA output (apartments, three districts) 47
Table 8: Annual Growth/Decline Rates Used in the Scenario Simulator (2019–2025) 58
Table 9: Initial Ensemble Model Performance (All Property Types) 59
Table 10: Comparative Performance of LightGBM vs. XGBoost for Apartments 60
Table 11: Final XGBoost Model Performance for Apartments 61
Table 12: Comparative Performance of LSTM vs. XGBoost for transaction forecasting 62
Table 13: Final XGBoost Model Performance for Transaction Forecasting (2016 Test Year) 63
Table 14: TimeGAN Performance Metrics (Successful Per-City Model) 70

Chapter 1: Introduction

1.1) Background

The global real estate industry, a cornerstone of economic stability, is undergoing a profound digital transformation. A sector, once dependent on traditional practices and expert intuition to one increasingly driven by data and technology 1][6]. At the forefront of this transformation is Artificial Intelligence (AI), which JLL's 2023 survey ranked among the top three technologies expected to reshape the industry, signaling its transformative potential for investors, developers, and occupiers alike [2].

AI and machine learning are becoming indispensable tools. They are revolutionizing core functions such as property valuation, market forecasting, assistance, and investment strategy by extracting actionable insights from vast and complex datasets covering market dynamics, property attributes, and consumer behavior [4] [2][5]. This marks a definitive pivot from reliance on heuristic, intuition-based assessments toward robust, data-driven decision-making, enhancing accuracy, operational efficiency, and strategic foresight while minimizing uncertainty [4][6][3].

This technological shift has been fueled by the proliferation of online real estate portals, such as Zillow, Trulia, and Realtor.com. By aggregating millions of listings, these platforms have created unprecedented public repositories of property data, moving the industry beyond its traditional reliance on localized, manual processes [7]. These platforms provide essential feedstock for AI-driven analysis by offering rich detail on property features, transactions, and emerging trends [7][5].

Yet, this explosion in data quantity has unveiled a critical and pervasive challenge: a fundamental deficit in data quality. Real estate data is notoriously disseminated, spread across hundreds of disparate regional Multiple Listing Services (MLSs), each with its own schemas, and standards. Even basic attributes like square footage or the number of bedrooms is inconsistently defined [9]. This is severely compounded by manual data entry errors from millions of individual agents and, in some cases, intentionally deliberate manipulation to bypass system rules. The result is a data ecosystem rife with disparity, inaccuracies, and unreliability [9].

Consequently, a central tension defines the modern data-driven approach: while the potential of AI is vast, the effectiveness of any analytical model is fundamentally determined by the quality of its underlying data [8]. This critical gap between the theoretical promise of big data and the messy reality of its application creates significant, and often underestimated, obstacles to robust and reliable analysis. Within this context, tools such as trend visualization and chat assistants become essential, translating complex analytics into accessible insights for both professionals and consumers.

1.2) Problem and motivation

While these challenges are recognized globally, they are magnified in emerging markets such as Lebanon. Unlike in the U.S. or Europe, where institutional datasets and standardized MLS systems exist, Lebanon lacks any centralized, accessible record of property transactions. Researchers and practitioners are therefore forced to rely on fragmented, web-scraped data from private portals sources that inherit the same global challenges of inconsistency, but without the counterbalance of scale. It is within this unique and difficult context that this project is situated.

This project was created to develop a comprehensive real estate analysis tool for the Lebanese market, leveraging publicly available, web-scraped data from local portals as a necessary alternative to inaccessible official records. While these sources provide a critical window into the market, the data itself presents two distinct and severe challenges. These challenges fundamentally dictated the project's methodological approach and defined its core research contribution: **how to deliver robust analytics in a context of extreme data scarcity and noise.**

1.2.1. The Time-Series Challenge: Forecasting on a Knife's Edge

The first fundamental constraint arose from the historical transaction data (transactions.csv). For a time-series forecasting task, the dataset is critically small, comprising a mere **360 rows** representing 72 months of activity across five key regions. This limited volume immediately rules out complex, data-hungry models which are prone to overfitting on such a short timeline.

Beyond its size, the dataset is characterized by high noise and significant month-to-month variance. Jagged, spiky movements dominate, obscuring the underlying yearly seasonal pattern. This volatility creates a substantial risk of models learning random fluctuations instead of genuine market signals. Consequently, this data reality mandated a conservative modeling strategy focused exclusively on preventing overfitting, employing robust algorithms like XGBoost with strongly regularized hyperparameters to force the model to prioritize broad seasonal and trend signals while ignoring noise.

1.2.2. The Tabular Data Challenge: Navigating a Landscape of Data Deserts

The second, more defining challenge arose from the property listings data (properties.csv). The issue was not merely one of imbalance but of severe fragmentation and profound regional sparsity, creating veritable "data deserts" for entire categories of assets.

A granular breakdown of the initial 2,849 listings, detailed in Table 1.1, reveals this stark reality.

Property Type	Beirut	Mount Lebanon	North Lebanon	Total
Apartments	**258**	**1,190**	**56**	**1,504**
House/Villa	0	136	21	157
Land	17	644	71	732
Chalets	1	114	1	116
Shops	23	98	9	130
Commercial Buildings	0	25	4	29
Offices	30	93	6	129
Residential Buildings	6	36	1	43
Factories	0	6	0	6
Gas Stations	0	2	0	2
Restaurants	0	1	0	1
Total	**335**	**2,345**	**169**	**2,849**

Table 1: Distribution of property listings by type and region

The table illustrates extreme class imbalance and regional sparsity. The "Apartments" category was the only segment with sufficient multi-regional coverage for viable model development. As its data shows, a unified, generalized price estimation model was fundamentally unfeasible. Key findings include:

- **Viable Category:** Only **Apartments** (1,504 listings) had meaningful representation across Beirut, Mount Lebanon, and North Lebanon.
- **Data Deserts:** The **House/Villa** category was completely absent (0 listings) in Beirut, the nation's economic capital.
- **Extreme Concentration: Land** listings were overwhelmingly concentrated in a single province, with 88% of all listings found in Mount Lebanon.
- **Niche Categories:** Other commercial and residential types were statistically insignificant, with totals often in the single digits.

This empirical reality forced a strategic, data-driven pivot. The project scope was deliberately and rigorously narrowed to focus exclusively on developing a robust price estimation model for apartments. Far from a limitation of ambition, this was a necessary methodological response to the dataset's structural deficiencies.

Together, these two challenges, a minuscule, noisy time-series and a fragmented, incomplete cross-sectional dataset– frame the central problem this thesis tackles: How to pioneer an analytical approach that overcomes severe data limitations to provide actionable real estate intelligence in emerging markets.

1.3) Research Questions and Objectives

To address these challenges of noise, sparsity, and fragmentation, this research is guided by three core questions:

RQ1: Data Quality and Model Performance
What level of predictive performance is achievable for real estate price estimation, trend visualization, and transaction forecasting when using small, web-scraped datasets characterized by significant noise and outliers?

RQ2: Synthetic Data Utility
How effective is TimeGAN in generating synthetic real estate transaction data that preserves the statistical properties of the original data and can be used to power a realistic market scenario simulator?

RQ3: Privacy and Data Augmentation
In what ways can synthetic data generation serve as a privacy-preserving alternative for sharing real estate market dynamics and augmenting limited datasets for analysis, while also powering features such as the chat assistant?

1.4) Project Overview and Thesis Statement

In direct response to the profound data challenges inherent to the Lebanese market, this research culminates in the development of a tangible, web-based solution: the Real Estate AI platform, a unified, web-based analytical tool designed to bring data transparency and AI-driven decision support to an otherwise opaque market.

The platform is architected to target two key audiences:

- **Homebuyers**: A data-driven price estimator (XGBoost model), trend visualization, chat assistance, and a Comparative Market Analysis (CMA) dashboard. Together, they provide realistic valuations (e.g., \$344,400 with a range of \$310,000–\$378,900) contextualized by district-level median prices and inventory.

- **Investors and Analysts**: An innovative Scenario Simulator, pivoting from fragile single-point forecasts to dynamic explorations of plausible "Boom" and "Crash" trajectories. Powered by five region-specific TimeGAN models and informed by economic narratives, the simulator transforms uncertainty into a tool for stress-testing investments.
  Crucially, its reliance on synthetic data ensures complete privacy, avoiding exposure of sensitive transaction records.

This thesis argues that while direct application of AI models can yield viable predictive outcomes from small and noisy real estate datasets, achieving advanced analytical capabilities like dynamic market simulation requires a strategic pivot to synthetically generated data to overcome the inherent limitations of the source information.

The remainder of this report is organized as follows: Chapter 2 reviews existing literature; Chapter 3 details the methodology; Chapter 4 presents results; Chapter 5 discusses implications; and Chapter 6 concludes with contributions and directions for future research.

Chapter 2 : State Of The Art

2.1) AI Applications in Real Estate: Prediction and Valuation

2.1.1. from traditional to modern valuation

Real estate valuation has long been acknowledged as a cornerstone of economic activity, shaping decisions made by households, investors, financial institutions, and policymakers.

For decades, valuation relied heavily on traditional statistical techniques, most prominently hedonic pricing models [10]. These models treat property prices as a linear function of observable attributes, such as size, number of rooms, location, and age of the property, providing a straightforward and interpretable framework.

However, while hedonic models played a foundational role in bringing systematic rigor to real estate valuation, they are ultimately constrained by their underlying assumptions. They presuppose that each characteristic of a property contributes to price in a fixed, additive way, and they often fail to capture the more subtle, non-linear relationships and spatial heterogeneity that characterize real estate markets [14], [11]. The limitations of these traditional approaches have become increasingly evident as real estate markets have grown more complex. A single attribute in isolation rarely determines prices; instead, they emerge from the dynamic interplay of multiple factors, including neighborhood quality, local amenities, economic conditions, and broader macroeconomic cycles.

Moreover, housing markets are inherently heterogeneous: two seemingly identical properties can diverge significantly in price due to subjective perceptions such as curb appeal or neighborhood prestige. Linear models, designed to impose order on this complexity, often oversimplify, resulting in valuation inaccuracies, particularly in volatile or highly segmented markets.

This recognition has driven the field toward machine learning (ML), which does not rely on pre-specified assumptions about functional form. Instead, ML models learn patterns directly from the data, enabling them to capture complex, non-linear interactions and to adapt flexibly to heterogeneous datasets. Research consistently shows that tree-based ensemble methods such as Random Forest, Gradient Boosting Machines, and especially XGBoost outperform traditional regression in predictive accuracy, robustness, and generalizability [15], [4]. Neural networks, in turn, provide even greater flexibility, though often at the expense of interpretability. Collectively, these approaches represent a paradigm shift from static, assumption-driven valuation toward data-driven intelligence that learns from the market itself.

The practical impact of this shift is visible in the real estate industry. Platforms such as Zillow have pioneered large-scale applications of ML through their *Zestimate* tool, which integrates structured data from tax records and MLS feeds with advanced models, including neural networks and decision trees. Although the tool has faced criticism for accuracy in certain markets, it nevertheless transformed consumer expectations by making automated valuation accessible to millions of users [16], [17]. Similar innovations can be seen in platforms like Redfin and Right move, which increasingly rely on AI not just for valuation, but also for recommendation systems, fraud detection, and personalized search.

Looking forward, the frontier of valuation research and practice lies in multimodal machine learning, where structured tabular data are combined with unstructured inputs such as text descriptions, satellite imagery, and even photographs of interiors and exteriors. This new wave of models has already demonstrated the ability to capture subtle cues, such as architectural style, finishing quality, or neighborhood greenery, that traditional models could never quantify [12], [18]. This transition from linear regression to adaptive, multimodal AI frameworks represents a fundamental redefinition of how real estate value is understood, predicted, and communicated. However, the effective application of these advanced techniques, particularly in markets characterized by data scarcity and high volatility, remains a significant challenge that this research seeks to address.

A cornerstone of traditional property valuation is the Comparative Market Analysis (CMA), a practical method widely used by real estate professionals to estimate property values based on the prices of recently sold comparable properties [19][20]. Underpinning the CMA is the principle of hedonic pricing, which treats a property's value as an aggregation of its features such as location, size, and condition. In this project, a simplified CMA framework was implemented using SQL queries and Python scripts to extract and aggregate transaction records, providing a transparent and reproducible baseline that mirrors industry practice.

However, CMA, like the hedonic models it relates to, faces inherent limitations. It depends heavily on the availability of suitable comparables and struggles to capture non-linear feature interactions, temporal dynamics, and broader macroeconomic [21]. These weaknesses are acutely exposed in regions with sparse data, such as the Lebanese market outside Beirut, where missing or noisy records often lead to non-robust results.

Therefore, while serving as a useful and interpretable benchmark, the CMA's limitations provide the direct rationale for this project's comparison with more advanced methods, including Automated Valuation Models (AVMs), XGBoost regressors, and generative approaches like TimeGAN. To enhance accessibility and interpretability, this project also integrates trend visualization to illustrate market dynamics over time and a chat assistant to guide users through valuation outputs, CMA comparisons, and predictive insights in an intuitive, conversational format.

2.1.2. Price Estimation Models

Building upon the general shift from traditional to machine learning (ML) methods, the application of ML to property price estimation has become a dominant paradigm, evidenced by both large-scale industry platforms and rigorous academic research. These approaches consistently demonstrate the ability to capture the complex, non-linear relationships that determine real estate value.

Industry Implementation: The Case of Zillow's Zestimate

The most notable example of ML-powered valuation is Zillow's Zestimate, an automated valuation model (AVM) that has become a household name. Initially launched using simpler models, the current Zestimate employs a sophisticated neural network that analyses hundreds of data points for over 100 million homes, including property characteristics from public records, multiple listing service (MLS) data, and market trends [22].

A marked innovation is its use of computer vision to analyze property photos, allowing the model to infer value-signaling features like the quality of finishes or curb appeal that are not captured in structured data alone [23]. While a groundbreaking tool for consumer transparency, the Zestimate's documented median error rate, approximately 2.4% for on-market homes and 7.9% for off-market homes [24], highlights the inherent challenges of mass valuation and sets a realistic benchmark for the field.

Academic Validation: The Superiority of Ensemble Methods

Academic literature provides robust validation for the industry's turn to ML.

Comparative studies across diverse markets have established that advanced tree-based ensemble methods, such as Random Forest and Gradient Boosting, significantly outperform traditional hedonic regression models in predictive accuracy.

Among these, the XGBoost algorithm has repeatedly emerged as a top performer due to its efficiency, handling of complex data relationships, and robustness against overfitting [25].

For instance, research on the Ames housing dataset found that XGBoost achieved superior results compared to support vector machines (SVM) and neural networks, while studies of markets in Boston and Melbourne have consistently shown ensemble methods delivering lower error metrics (RMSE, MAE) and higher explanatory power (R^2) than linear models [26]. This body of work confirms that the ability of these models to automatically learn interaction effects and non-linear patterns is key to their success.

Application in This Research: A Pragmatic Approach

Guided by this academic consensus, the price estimation component of this research employed XGBoost, selected for its proven performance and suitability for the project's specific data constraints.

The model was trained on a curated dataset of apartment listings, the only property type with sufficient cross-regional data, after stringent preprocessing to handle outliers, null values, and regional imbalances. Strategic feature engineering, including the creation of location-size interactions and space efficiency ratios, provided the model with critical signals. The result was a performant and practical model, achieving a viable R^2 of 0.78 and a Mean Absolute Error (MAE) of $88,475, a substantial improvement over an initial, failed attempt to build a unified model for all property types ($R^2 = 0.2$).

In conclusion, the trajectory of price estimation is firmly landed in machine learning, with industry leaders like Zillow setting the scale and academic research validating the superiority of ensemble methods like XGBoost.

This project aligns with and reinforces this trajectory, demonstrating that a focused, well-engineered application of these techniques can yield accurate and actionable insights even within the challenges of a data-scarce emerging market.

2.1.3. Transaction and Market Forecasting

Forecasting real estate market trends and transaction volumes presents a distinct set of challenges compared to static valuation, primarily due to the time-dependent nature of the data. The evolution of forecasting methodologies mirrors that of valuation: a journey from classical statistical models toward sophisticated machine learning techniques.

However, a critical lesson from forecasting research is that model complexity does not automatically guarantee superior accuracy. In fact, on small, noisy datasets, a common reality in real estate, simpler, well-regularized models often outperform complex ones that are prone to overfitting [27] [28].

In contemporary practice, two dominant approaches have emerged for real estate forecasting:

(1) Classical Time-Series Models: Such as ARIMA (AutoRegressive Integrated Moving Average) and its multivariate extensions (VAR), which model future values based on past values and errors.
(2) Supervised Machine Learning Models: Where the forecasting problem is reframed into a supervised learning task. Techniques like gradient-boosted trees (e.g., XGBoost, LightGBM) are applied to a feature-engineered table containing temporal features like lags, rolling statistics, and encoded seasonality (e.g., month, quarter).

A growing consensus in the literature suggests that for tabular, feature-engineered time-series data, tree-based ensembles are exceptionally competitive. They often rival or surpass more complex neural network architectures, offering the added benefits of faster training times, lower computational cost, and greater interpretability through feature importance scores [27], [28].

This is particularly true in contexts characterized by modestly sized, noisy datasets, a perfect description of most web-scraped real estate transaction histories.

While deep learning models like Long Short-Term Memory networks (LSTMs) excel in domains with long, rich sequential data (e.g., NLP, high-frequency finance), their application to short, volatile real estate series is fraught with difficulty.

Studies have shown that LSTMs can struggle with the outlier-prone, "spiky" nature of monthly transaction data, a challenge noted even in research reporting otherwise strong results (Open-access study on Saudi housing prices). Their high capacity makes them susceptible to learning noise rather than signal when data is limited.

This project's dataset, comprising only 72 monthly observations across five regions, with significant noise obscuring a clear seasonal signal, is a textbook example of the conditions where robust, regularized models are advised. Guided by this literature, the forecasting approach was deliberately designed to prioritize generalization over complexity. The problem was framed as a supervised learning task for XGBoost, with heavy use of regularization (e.g., max_depth=3, low learning_rate) and carefully constructed temporal features (e.g., cyclical encoding of months, lagged values, binary flags for seasonal peaks) to help the model isolate the underlying trend from stochastic volatility.

The empirical results strongly validated this methodological choice. An initial LSTM model catastrophically overfitted the training data, producing unusable forecasts with a high Mean

Absolute Error (**MAE: 127.01**) and Root Mean Squared Error (**RMSE: 176.10**). In contrast, the tuned XGBoost model successfully learned the core seasonal pattern while remaining robust to noise, achieving a viable test-set Mean Absolute Percentage Error (**MAPE: 22%**). This outcome aligns perfectly with the broader finding: in data-scarce environments, a simpler, well-specified model is vastly superior to a complex one that overfits.

Finally, the field is advancing towards hybrid frameworks that integrate ML with traditional statistical insight. The most promising approaches combine the pattern-recognition power of machines with an understanding of macroeconomic shocks and regime changes. This project's strategic pivot from a pure prediction task to a **scenario simulation engine**, using synthetic data to model "Boom" and "Crash" trajectories, is a direct contribution to this frontier. It acknowledges that in highly uncertain markets, providing a range of plausible futures based on expert-defined growth rates is more valuable and resilient than a single, fragile point forecast [29].

2.1.4. AI Chat Assistant (Conversational Search)

The real estate industry has historically relied on manual data collection, expert intuition, and localized decision-making. With the rise of digital platforms such as Zillow, Trulia, and Realtor.com, vast amounts of property data have become publicly available, enabling AI-driven valuation and trend analysis at scale. However, the sheer volume and fragmentation of this data create challenges for end-users, who often struggle to interpret raw outputs or navigate complex dashboards.

The AI chat assistant addresses this gap by providing a conversational interface that allows users to query the system in natural language. Instead of manually filtering datasets or interpreting graphs, users can ask questions such as “What are the price trends in Beirut?” or “Compare property prices between Tripoli and Sidon.” The assistant interprets intent, retrieves relevant outputs, and delivers structured, context-aware responses. This functionality is particularly valuable in markets like Lebanon, where data scarcity and inconsistency amplify the need for accessible, guided interpretation.

The system is connected to **OpenRouter** with cost-efficient model such as **meta-llama** and ensuring scalability without quota limitations. A key technical challenge was maintaining conversational continuity: early iterations failed to persist chat history, but this was resolved through session-based storage, allowing users to refine queries iteratively and build on prior interactions.

By embedding conversational AI into the dashboard, the project aligns with the broader industry trend of using machine learning not only for prediction but also for **decision support and accessibility**. The assistant democratizes access to real estate intelligence, making advanced analytics usable for both professionals and non-technical stakeholders.

2.1.5. Trend Visualization (City Circles + Price Trends)

The proliferation of digital real estate platforms has made millions of listings and transactions publicly available, creating unprecedented opportunities for AI-driven valuation and forecasting. Yet, as research highlights, the effectiveness of these models is constrained by fragmented datasets, inconsistent attribute definitions, and frequent human input errors. In such contexts, visualization becomes critical: it translates noisy, complex data into interpretable signals that can guide decision-making.

The trend visualization module addresses this need by combining **City Circles** with **City Price Trends** on a map. City Circles represent each city as a circle whose size corresponds to market magnitude (e.g., transaction volume or average price) and whose color encodes directional trends. Hovering over a circle reveals a tooltip with the city name, trend status, and percentage change.

Complementing the spatial view, the **City Price Trends** visualization presents arrows showing if prices are "Increasing," "Decreasing," or "Stable" based on comparing median price for each year to describe market direction . It allows users to compare cities over time, identify long-term growth or decline patterns, and quantify percentage changes.

By situating these visualizations within the broader context of fragmented and noisy real estate datasets, the project highlights their role as **bridges between raw data and actionable insight**. Policymakers can identify regional disparities, investors can spot emerging hotspots, and developers can track long-term momentum. It bridges the gap between raw data and strategic insight, enabling users to **interpret market dynamics** at a glance while retaining the ability to drill down into city-level detail.

2.2) The Pervasive Challenge: Data Scarcity and Quality in Real Estate Analytics

2.2.1. The "Data-Rich, Information-Poor" Problem

The expansion of online real estate portals creates the illusion of abundant, readily available data for analytics. In practice, however, this web-scraped data is often characterized by a fundamental paradox: it is simultaneously vast in volume and poor in reliable information, a classic **"data-rich, information-poor" (DRIP)** environment.

Public listings are often incomplete, with missing critical attributes such as square footage, number of bedrooms, or transaction history. They are also prone to inconsistency, with varying formats for locations, mixed units of measurement, and a high degree of duplication. Furthermore, the data can be intentionally misleading, as agents may inflate descriptions or create duplicate listings with slight variations to game platform algorithms and attract buyer inquiries. These challenges are well-documented in industry and technical literature, which highlight the significant efforts required in data cleaning, normalization, and validation before scraped data becomes viable for analysis. 30][31]

This quality deficit is not confined to emerging or niche markets. Even industry leaders like Zillow explicitly state that the accuracy of their flagship Zestimate® is directly bound by the quality and availability of underlying data from county records and Multiple Listing Services (MLSs). Their reported median error rates are consistently lower for on-market homes (where data is fresh and abundant) compared to off-market homes (where data may be stale or sparse), underscoring that platform scale alone cannot overcome inherent data limitations [32], [17].

This presents a global challenge: the effectiveness of any data-driven model is ultimately constrained by the integrity of its input data.

The response to this challenge varies by market structure. In China, for example, dominant platforms like Lianjia (Beike) have adopted a highly centralized, institutional approach to data governance, creating standardized housing dictionaries and vetting agent networks to enforce data quality and reduce duplication at the source. Despite this top-down effort, academic research continues to document issues with inconsistent transaction recording and data fragmentation across multiple Chinese platforms and regions, proving that data quality is a persistent and universal hurdle. 33][34]

For this research, which focuses on the Lebanese market, these global patterns were directly observed. The initial web-scraped dataset, while substantial in row count, required extensive preprocessing, including canonical normalization of regional names, deduplication, and coercion of mixed data types to become a usable foundation for modelling.

This experience confirms a critical axiom for real estate analytics: quantity without quality is insufficient; clean, consistent, and well-documented data is the non-negotiable prerequisite for any reliable analytical outcome.

2.2.2. Common Data Issues: Sparsity, Noise, Imbalance, and Historical Depth

The overarching "data-rich, information-poor" dilemma manifests in several specific, well-documented challenges that critically undermine the reliability of real estate analytics: sparsity, noise, regional imbalance, and a lack of historical depth.

Sparsity and missing values

A fundamental issue with web-scraped data is the pervasive absence of critical attributes. Listings frequently lack essential details such as the number of bedrooms, square footage, or even the price itself.

In some markets, the problem is structural; an analysis of Dubai property portals found that crucial fields like rental value were missing in over 90% of records, rendering vast portions of a dataset unusable for modelling without extensive imputation or filtering [35].

Academic research actively seeks solutions to this problem, such as using multimodal deep learning to predict missing housing attributes from images, which demonstrates the lengths required to overcome inherent data sparsity [36].

Noise and inaccuracy

Beyond missing data, the available information is often degraded by noise and inaccuracies. Listing prices can be intentionally inflated, descriptions can contain exaggerations or typos, and human errors during data entry are common.

This results in a low signal-to-noise ratio that can severely distort model training, leading to unreliable and biased predictions. The literature on machine learning consistently identifies robust data cleaning and outlier detection, using methods like Isolation Forest or interquartile range (IQR) filtering, as a non-negotiable first step to ensure model integrity in real estate applications [37].

Regional imbalance

Real estate data is notoriously geographically skewed. Scraping efforts naturally yield a high density of listings from major urban centers, while rural and peripheral regions become "data deserts," effectively invisible to analytical models. This imbalance reflects and reinforces existing spatial inequalities in economic development and infrastructure [38].

Academic studies using network analysis have shown that real estate data naturally cluster into spatially segregated submarkets, even within a single country [39].

This phenomenon was directly observed in this project's dataset, where the '**Apartment**' category was the only type with sufficient cross-regional representation (Beirut, Mount Lebanon, North Lebanon) to support a viable model, while others like '**House/Villa**' were absent entirely from Beirut, forcing a strategic narrowing of the project's scope.

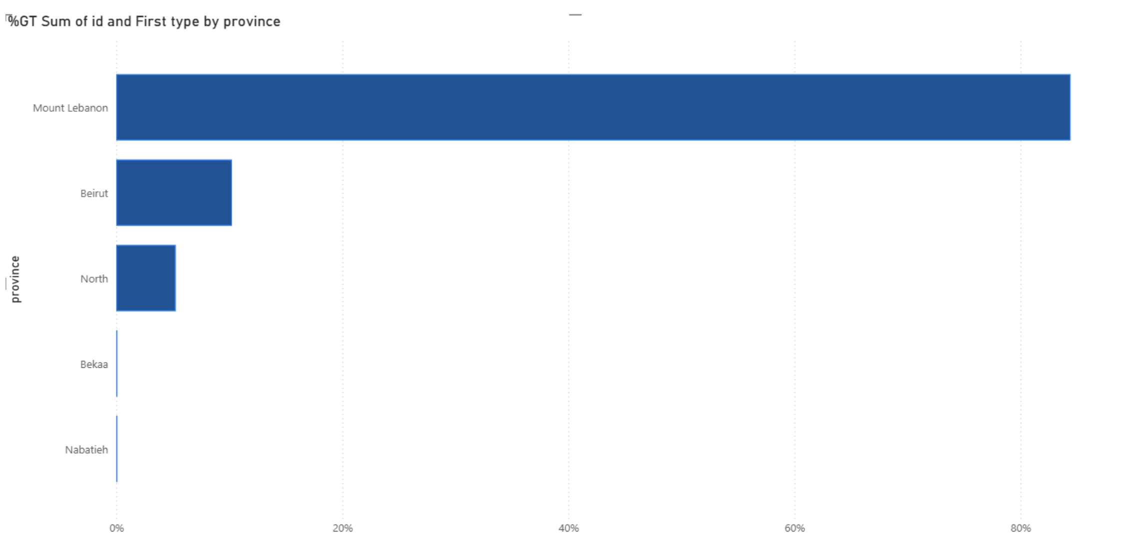


Figure 1: Regional Distribution of Apartment Listings

In figure, the data reveals a pronounced geographic imbalance, with the majority of listings (84.4%) concentrated in Mount Lebanon, followed by Beirut (10.2%) and the North (5.3%). The provinces of Bekaa and Nabatieh are unrepresented, illustrating the data scarcity in peripheral regions that characterizes many emerging real estate markets

Lack of Historical Depth

Finally, reliable time-series forecasting requires long, consistent historical data.

Web-scraped datasets typically offer only a recent snapshot, spanning months rather than the years or decades needed to train robust models that can understand long-term trends and cycles. This scarcity of temporal data is a severe constraint. Advanced research attempts to compensate for this through sophisticated statistical methods, such as building hyperlocal price indices by pooling data across zones using Bayesian dynamical models [40]. When such pooling is not possible, the field must look to other solutions.

This fundamental limitation of raw historical data provides a strong rationale for the strategic pivot employed in this project: using synthetic data generation via TimeGAN to augment a short time-series and enable scenario-based simulation where traditional forecasting would be unreliable.

2.2.3. The Privacy Conundrum

Even when real estate data is technically accessible and structured, its practical use is severely constrained by escalating privacy regulations and ethical considerations.

Property transactions are inherently sensitive, containing personally identifiable information (PII) such as names, financial details, and precise geolocation data. This triggers compliance with stringent global data protection laws. For instance, the European Union’s General Data Protection Regulation (GDPR) explicitly classifies geolocation data as sensitive, requiring robust anonymization for lawful processing (European Commission, GDPR). This regulatory landscape is mirrored in the Gulf region, where frameworks like the UAE's PDPL (2022), Saudi Arabia's PDPL (2021, updated 2023), and Qatar's Privacy Law (2016) mandate explicit user consent, restrict cross-border data transfers, and impose severe penalties for violations, creating a complex compliance environment for real estate data handling. [41][42]43][44]

These legal requirements intersect with competitive market practices. Industry analyses, such as those from JLL (2023)[45], note that firms increasingly treat proprietary data as a core competitive asset, further restricting open access to reliable transaction records for external analysis. This commercial reality exacerbates the data scarcity problem. Furthermore, as Clara [4] underscores, the adoption of AI in real estate introduces critical **"ethical concerns, regulatory gaps, and data bias,"** emphasizing that data security and algorithmic fairness are non-negotiable prerequisites for building trustworthy systems.

In direct response to this dual challenge of privacy and scarcity, **synthetic data generation emerges as a compelling and practical solution**. This approach involves creating entirely artificial datasets that replicate the statistical properties and multivariate relationships of the original data without containing any actual PII.

Research from institutions like MIT demonstrates that such synthetic data "looks like [real data]… but doesn’t contain or even hint at any of the information from the original data" [46]. This method offers superior protection compared to traditional anonymization, which remains vulnerable to re-identification attacks. The National Institute of Standards and Technology (NIST) affirms that techniques like differential privacy can provide "provable privacy guarantees" for synthetic data, enabling secure and compliant analysis [47]. This is validated by industry leaders. Google Cloud, for example, showcases how synthetic data generated via tools like BigQuery can facilitate collaboration and innovation without compromising privacy [48].

For this project, these considerations are central, not peripheral. In the Lebanese context, where transaction data is already scarce and shrouded in opacity, adhering to these global privacy standards is imperative. The strategic decision to employ synthetic data, particularly for the public-facing scenario simulator, directly addresses this conundrum. It transforms a significant limitation, data confidentiality, into a core design strength, enabling powerful modelling and simulation within a legally compliant and ethically sound framework.

Notably, Lebanon operates in a regulatory vacuum, lacking a comprehensive data protection law equivalent to the GDPR or GCC PDPLs. This legislative gap, a consequence of prolonged political and economic instability, leaves real estate data collection and sharing in a legal grey zone.

In this context, the choice to use synthetic data is not merely technical but **ethically imperative**. By adopting a privacy-by-design approach that exceeds local legal requirements, this project proactively mitigates the risk of mishandling sensitive financial and location data. This strategy builds inherent trust and ensures the framework is both ethically sound and future-proof, establishing a higher standard for responsible data analytics in Lebanon's emerging proptech landscape.

2.3) Synthetic Data Generation as a Strategic Solution

2.3.1. Introduction to Generative Adversarial Networks (GANs)

Encountering the inherent limitations of small, noisy, and privacy-constrained datasets, this research turns to a cutting-edge solution: **Synthetic data generation via Generative Adversarial Networks (GANs)**.

Introduced by Goodfellow et al. in 2014 [49], GANs are a class of AI frameworks designed not only to analyze data, but to **generate entirely new, synthetic samples** that capture the underlying statistical distribution of the original data.

The core innovation of a GAN is its adversarial training process, which pits two neural networks against each other in a minimax game: a **Generator** and a **Discriminator**. This hostile relationship is the key to a GAN's ability to produce highly realistic outputs.

1. The **Generator** creates synthetic data from random noise, starting with random noise and attempting to transform it into something that resembles the real data.
2. The **Discriminator** acts as a judge, trained on real data to distinguish between authentic samples and the fake data produced by the generator.

   These two networks are locked in a "zero-sum game."

Through this competition, the Generator is compelled to produce increasingly realistic data until the Discriminator can no longer tell the difference. This process effectively teaches the model to internalize and replicate the complex patterns, correlations, and variances present in the training set.

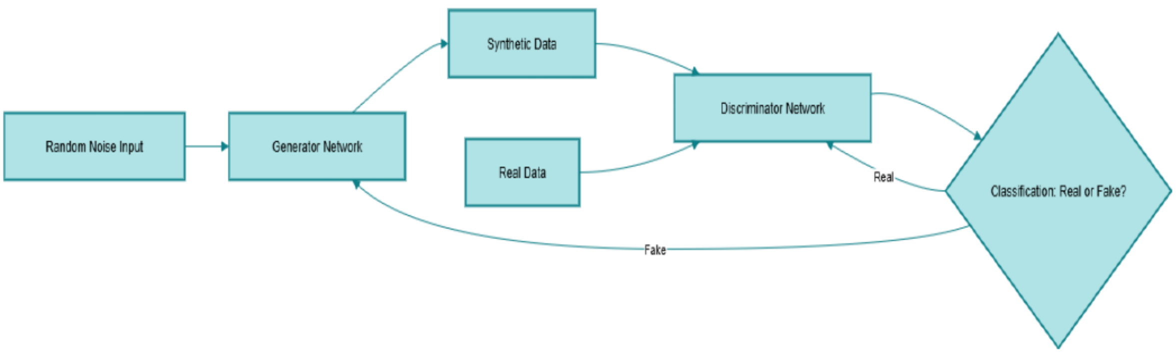


Figure 2: Generalized Architecture of a Generative Adversarial Network (GAN).

While GANs first gained fame for generating images (e.g., generating photorealistic human faces or creative artwork [50]), their architecture is highly adaptable.

Subsequent research has developed specialized GANs for diverse data types, including tabular data (e.g., CTGAN, which handles diverse data types, skewed distributions, and class imbalances [51]) and, most critically for this project, **time-series data.**

This versatility makes GANs a powerful strategic tool to overcome data scarcity, balance imbalanced classes, and create privacy-preserving datasets for robust real estate analytics.

2.3.2. GANs for Tabular and Time-Series Data

While GANs revolutionized image generation, their application to structured data required significant architectural innovation. Tabular and time-series data present unique challenges, including mixed data types (continuous, categorical) and, crucially, the need to preserve **temporal dependencie**s; the sequential relationships that are the essence of time-series analysis.

Specialized models like the Conditional Tabular GAN (CTGAN) were developed to address the complexities of static tabular data. CTGAN introduces techniques like conditional sampling to handle imbalanced categorical variables and skewed distributions, significantly improving the statistical fidelity of synthetic outputs for tasks like property price estimation [52].

Subsequent variants like CTAB-GAN further enhance realism by incorporating auxiliary losses. However, a critical limitation emerged: these tabular GANs treat each row as an independent sample. They are designed to replicate the overall *distribution* of features but fail to capture the *temporal dynamics* between them. They can generate a realistic-looking snapshot of the market but cannot model how prices and transactions evolve over time.

This limitation was confirmed empirically in this project. An initial application of a tabular GAN to the transaction dataset successfully generated privacy-preserving data but resulted in low temporal fidelity.

The synthetic sequences failed to capture essential patterns such as regional seasonality and price trajectories, rendering them unsuitable for forecasting. This experience provided a key insight: **for time-series, privacy and utility are not mutually exclusive goals, but they require a specialized architecture**. This failure directly motivated the pivot to TimeGAN, an architecture designed explicitly for sequential data.

2.3.3. TimeGAN: The State of the Art for Sequential Data

The critical failure of tabular GANs to model temporal relationships necessitates an architecture designed explicitly for sequential data. TimeGAN (Time-series Generative Adversarial Networks), introduced in the seminal work by Yoon, Jarrett, and van der Schaar [53], represents this specialized breakthrough. Its novel design seamlessly integrates **unsupervised adversarial training** with **supervised temporal learning** to capture both the statistical distribution and the dynamic evolution of time-series data.

TimeGAN's superiority stems from its unique three-component architecture, which directly addresses the shortcomings of previous models:

1. Autoencoder (Embedding & Recovery Networks): Learns a compressed latent representation of the time-series data, filtering out noise while preserving the most salient features. This ensures the subsequent adversarial game is played in a meaningful, lower-dimensional space.

2. Supervisor Network: TimeGAN's pivotal innovation. Acting as an internal teacher, it learns temporal transitions inherent in the real data. It provides step-by-step guidance to the generator, enforcing the generation of sequences that follow plausible, realistic trajectories over time, something entirely missing in standard GANs.

3. Adversarial Network (Generator & Discriminator): Operating within the learned latent space, this classic GAN duo ensures the synthetic sequences are statistically

indistinguishable from real ones. The supervised guidance makes this adversarial process more stable and effective.

This hybrid architecture makes TimeGAN uniquely suited for challenging, real-world datasets like the one in this project: **small, noisy, and multi-dimensional time-series**. Where tabular GANs failed to capture the quarter-to-quarter evolution of regional transaction values, TimeGAN is specifically designed to learn and replicate these dynamics. Its ability to preserve **temporal coherence**, the most critical aspect of transaction data, is what enables the generation of a realistic synthetic baseline for the project's scenario simulator.

As such, TimeGAN was not merely a choice but the essential methodological solution for achieving advanced, privacy-preserving market analysis.

2.4) Web Technologies for Data-Intensive Applications

The success of large-scale real estate analytics platforms such **as Zillow in the United States** and **Lianjia in China** highlights the central role of web technologies in delivering advanced AI capabilities to end users.

Zillow's flagship tool, the **Zestimate**, relies not only on sophisticated machine learning models but also on a scalable web infrastructure that serves **hundreds of millions of monthly users** with near real-time price estimates and interactive analytics [54].

Similarly, Lianjia integrates **cloud-based big data pipelines and AI-driven recommendation engines** to provide highly personalized property search and valuation tools for Chinese markets. These examples demonstrate that the technical challenge is not only about building predictive models, but also about embedding them into resilient, data-intensive web systems.

Modern frameworks such as **React** (for responsive interfaces), **Quart/Django/Node.js** (for scalable backends), and **cloud-hosted relational databases** (e.g., PostgreSQL, Supabase) enable researchers and developers to integrate complex AI workflows into seamless applications. These stacks are designed to handle the dual challenge of **real-time responsiveness** and **large-volume data processing**, making them particularly suited for property analytics platforms.

In this project, a comparable architecture was adopted on a smaller but targeted scale: **Quart as backend, Supabase for data storage and security**, and **React Query for efficient front-end data fetching and state management.** By following industry practices while tailoring the system to Lebanon's fragmented property market, this project demonstrates how cutting-edge real estate intelligence can be made both accessible and context-specific.

Platform	Region	Key Features	Underlying Technologies
Zillow (Zestimate)	U.S.	Automated valuation model (AVM), interactive maps, transaction history, CMA dashboards	Large-scale ML models, cloud data pipelines, React/Node.js web stack, distributed databases
Lianjia (贝壳找房 / Beike)	China	AI-driven property search, price trends, neighborhood intelligence, personalized recommendations	Cloud-based big data systems, AI recommender engines, mobile-first apps
This Project	Lebanon	Price estimator, CMA dashboard, transaction forecasting, synthetic data simulator, chat assistant, trend visualization	Quart backend, Supabase database, React Vite frontend, XGBoost + TimeGAN models

Table 2: Comparative Overview of Real Estate Platforms

Having established the limitations of conventional approaches and the promise of advanced generative models, the next chapter will outline the methodological framework adopted in this study.

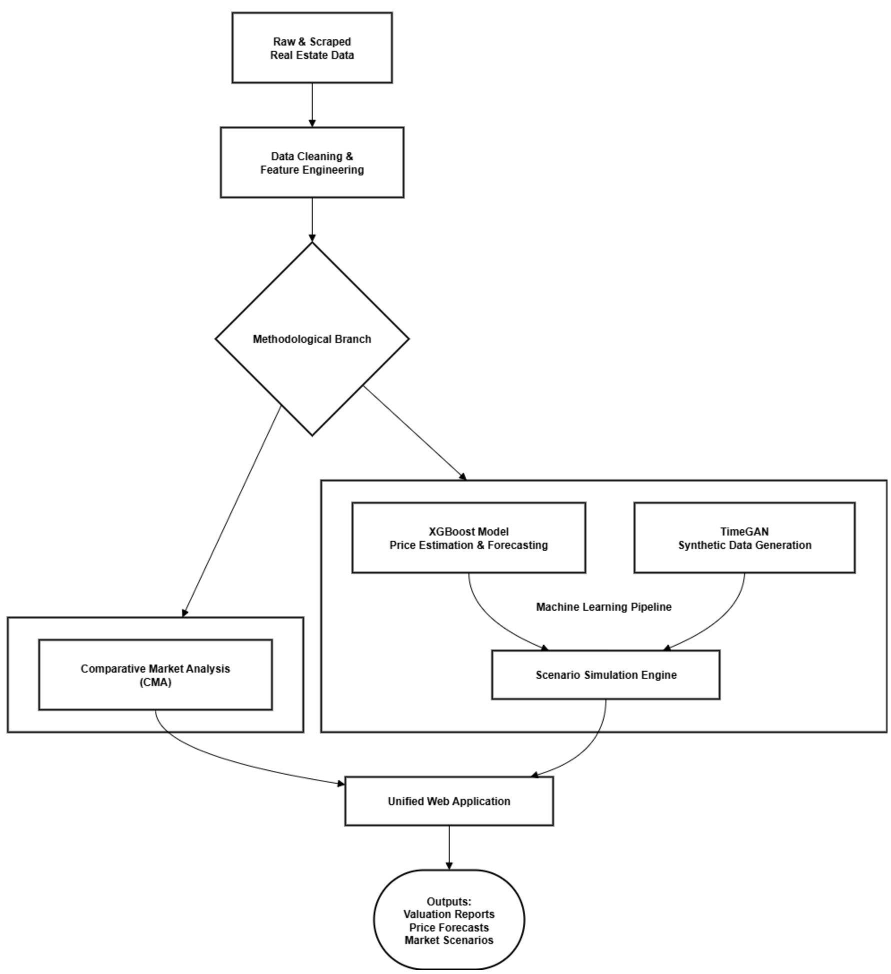


Figure 3: high-level overview of the modeling pipeline

The above figure highlights how raw transactional data is progressively transformed into predictive insights and scenario simulation.

Notably, this architecture also supports **interactive features** such as the **chat assistant**, which provides natural language access to complex analytics, and the **trend visualization module**, which translates noisy datasets into intuitive, real-time signals. These components exemplify how technical infrastructure and user-facing intelligence converge to democratize advanced real estate analytics.

The review of global platforms and the technical architecture adopted in this project underscores a central point: advanced real estate analytics is not only a matter of model sophistication but also of embedding those models within resilient, user-oriented systems. Having established the limitations of conventional data sources, the promise of generative models, and the enabling role of modern web stacks, the next step is to formalize the methodological framework. This framework details how synthetic data generation, predictive modelling, and interactive features are operationalized into a coherent, context-specific platform for Lebanon's real estate market.

Chapter 3: Methodology

This chapter presents the complete methodology employed to design and implement the AI-powered real estate web application. It begins by outlining the system's technical architecture and the technology stack underpinning the platform.

Next, it details the processes of data collection, database schema design, and feature engineering, which transform raw property and transaction data into structured, analyzable inputs.

The chapter then describes the implementation of both the traditional Comparative Market Analysis (CMA) tool and the machine learning models for price estimation, forecasting, chat assistance, price trend visualization and synthetic data generation.

Finally, it introduces the design and logic of the scenario simulator, the novel feature of this project that enables dynamic, volatility-aware market simulations.

3.1) System Architecture and Technology Stack

The "Real Estate" application is developed around a modern, scalable client–server architecture that ensures robust data handling, efficient model inference, and seamless user interaction.

At the frontend, a responsive React-based interface communicates asynchronously with a Python-based Quart backend via a RESTful API.

The backend handles both the execution of machine learning models and the traditional market analysis logic, with data persisted in a PostgreSQL database hosted on Supabase.

This modular architecture enforces a clear separation of concerns: the frontend manages interactivity and visualization, the backend focuses on data processing and inference, and the database ensures reliable storage and query efficiency. Such a layered design facilitates scalability, maintainability, and the integration of future analytical tools. Importantly, this architecture also supports **interactive features** such as the **chat assistant**, which provides natural language access to analytics, and the **trend visualization module**, which translates raw outputs into intuitive, real-time insights.

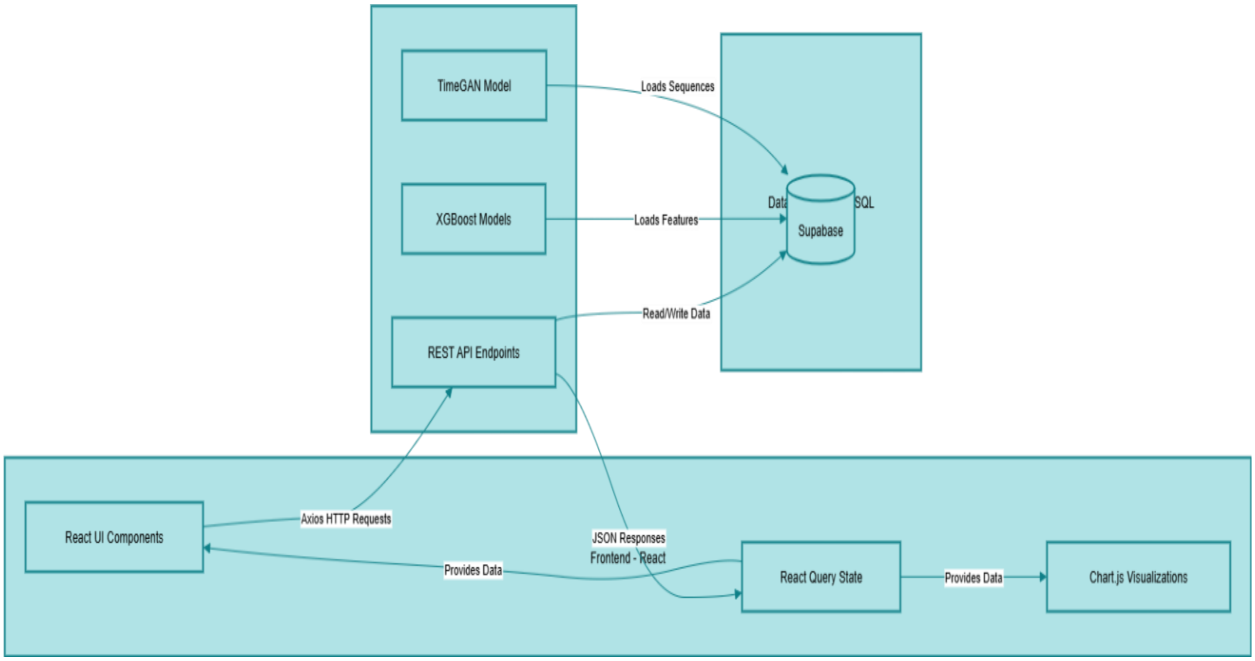


Figure 4: System Architecture Diagram

3.1.1. Technology Stack Breakdown

a) Frontend (Client):

- **React (Vite):** Core framework for building a modular, responsive, and performant user interface.
- **Axios:** Handles HTTP communication with the backend REST API.
- **React Query (@tanstack/react-query):** Manages asynchronous data fetching, caching, and synchronization, reducing redundant network requests.
- **React Chart.js 2:** Provides interactive data visualizations for forecasting, simulations, and CMA comparisons.
- **React Autosuggest:** Enhances the price estimator tool by providing real-time, location-based input suggestions.
- **Chat Assistant & Trend Visualization:** Integrated into the React layer to deliver conversational analytics and dynamic market signals.

b) Backend (Server):

a. **Python (Quart):** Asynchronous web framework supporting REST API endpoints and enabling scalable model inference.
b. **Pandas & NumPy:** Core libraries for numerical and tabular data manipulation.
c. **XGBoost:** Primary ML algorithm for price estimation and forecasting, optimized for speed and accuracy.
d. **Scikit-learn:** Provides preprocessing utilities (scaling, encoding), model selection methods, and evaluation metrics.

e. **TensorFlow/Keras:** Used exclusively for the deep learning component (TimeGAN model).
f. **Statsmodels:** Integrated for classical time-series baseline models (e.g., Holt-Winters Exponential Smoothing) to benchmark forecasting.
g. **Supabase-py:** Python client for secure database communication.
h. **Dotenv & Quart-Cors:** Ensure secure environment variable management and cross-origin resource sharing.

c) Database Layer:

- **PostgreSQL:** Robust relational database used for structured storage of properties, transactions, and user queries.
- **Supabase:** Backend-as-a-service platform hosting PostgreSQL, providing authentication, query functions, and efficient execution of SQL-based aggregations (e.g., market_comparison_query for CMA).

3.2) Data Collection and Schema

The foundation of the Real Estate application rests on two distinct datasets: **property listings and historical transaction records.**

These were acquired through a multi-pronged collection strategy tailored to the Lebanese market, where official, centralized, and digitized real estate data sources remain scarce. In the absence of national registries or public APIs, publicly available real estate portals became indispensable substitutes, while historical transaction data was sourced separately from a verified, confidential provider. This combination reflects both the opportunities and constraints of real estate analytics in Lebanon: valuable data exists but remains fragmented, unstandardized, and incomplete.

3.2.1. Web Scraping Process

The property listings dataset was originally compiled from two Lebanese real estate platforms, JSK Real Estate [56] and RealEstate Lebanon [57]. While I did not personally conduct the scraping, I was provided with both the cleaned dataset and the underlying Python scripts used to generate it.

Reviewing these scripts offered insight into the collection pipeline: for JSK Real Estate, listings were iteratively extracted from multiple HTML pages using the requests library for HTTP calls and BeautifulSoup for HTML parsing. Structured attributes such as address, price, property type, size, and number of bedrooms and bathrooms were parsed, with missing values handled explicitly (e.g., entries marked “N/A”).

For RealEstate Lebanon, data was accessed through authenticated API endpoints that returned listing data in JSON format, including attributes such as title, price, area (sqm), furnishing status, and geospatial identifiers (District, Province, Community).

Both scrapers incorporated short delays (time.sleep() of 1–2 seconds) between requests to prevent server overload and to follow ethical scraping practices. The outputs from both sources were subsequently consolidated into a unified CSV file, which I employ in this study in its processed form.

The transactions dataset was obtained from BRITE, a historical data provider affiliated with BLOMINVEST Bank [58], which compiles data from the General Directorate of Land Registry and Cadastre. This dataset was received in pre-aggregated form and excludes unregistered sales contracts. It contains the following variables: transaction ID, date, transaction number, city, and transaction value.

Together, these two datasets provide complementary perspectives on the Lebanese real estate market: the listings reflect current supply-side characteristics, while the transactions capture officially recorded market activity.

Although the scraping process itself was not carried out as part of my contribution, examining the code has informed my methodological understanding and will serve as a foundation for future applications of web scraping in related projects.

3.2.2. Database Schema

All collected data were stored in a PostgreSQL relational database hosted on Supabase, providing a managed, secure environment with support for SQL queries, persistence, and integrity checks.

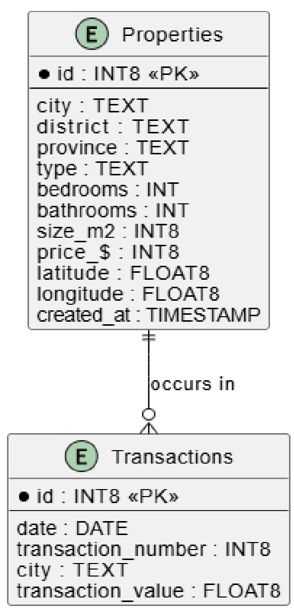


Figure 5: Entity-relationship diagram

The diagram that illustrates the structure and relationship between the two core tables that underpin the application's analytical tools.

3.2.3. A Peak in Database

a. Properties Table: This table stores individual property listings and serves as the primary data source for the **Price Estimation Model (XGBoost)** and the **Comparative Market Analysis (CMA) Tool**.

Column Name	Data Type	Description	Constraints
Id	INT8	Unique identifier for each listing.	PRIMARY KEY
City	TEXT	The city where the property is located.	-
district	TEXT	The district within the city.	-
province	TEXT	The broader province/region.	-
Type	TEXT	Type of property (e.g., Apartment, Villa).	-
bedrooms	TEXT	Number of bedrooms. Stored as text initially.	-
created_at	TIMESTAMP	Timestamp of when the record was created.	-
price_$	INT8	Listing price in U.S. dollars.	-
latitude	FLOAT8	Geographic latitude.	-
longitude	FLOAT8	Geographic longitude.	-
bathrooms	TEXT	Number of bathrooms. Stored as text initially.	-
size_m2	INT8	Size of the property in square meters.	-

Table 3: Properties Table Schema

Note: The *Properties* table contains **2,918 rows**. Several numerical columns (e.g., bedrooms, bathrooms) were initially scraped as TEXT, demanding type conversion during preprocessing.

b. Transaction Table: This table stores aggregated historical market activity and is the fundamental data source for the **Transaction Forecasting Model (XGBoost)** and the **Synthetic Data Augmentation (TimeGAN).**

Column Name	Data Type	Description	Constraints
Id	INT8	Unique identifier for each record.	PRIMARY KEY
Date	TEXT	The month and year of the transaction activity.	-
transaction_number	INT8	The volume of transactions recorded.	-
City	TEXT	The city where the transactions occurred.	-
transaction_value	FLOAT8	The aggregate value of transactions.	-

Table 4: Transaction Table Schema

Note: The *Transactions* table contains **360 rows** (72 months across 5 cities). The date and city columns required parsing and standardization for time-series analysis.

Despite their inherent structural limitations, sparsity, type inconsistencies, and missing values, these datasets provided the essential raw inputs for subsequent **feature engineering** and **AI model development**, forming the empirical backbone of the Real Estate application.

3.3) Data preparation, Feature Engineering and Model Implementation

To transform the raw, heterogeneous data into features suitable for the AI-driven Price Estimation model, a multi-stage pipeline was implemented. This pipeline addressed issues of data quality, structural inconsistencies, and the need to capture nuanced market signals through feature engineering.

3.3.1. General Cleaning

Upon initial collection, the raw data from their primary sources (property listings scraped from *JSK* [56] and *Real Estate* [57], as well as historical market activity from BLOMINVEST Bank [58]) underwent a universal cleaning pipeline to ensure consistency, reliability, and usability across all subsequent analytical tasks.

a) Initial Data Loading and Type Correction

The initial phase of data preparation involved loading the raw data into Pandas DataFrames, followed by a comprehensive cleaning process to convert text-based numerical values into their appropriate numerical formats. This included standardizing columns for price, property size, bedrooms, and bathrooms.

For prices, non-numeric characters such as currency symbols ($), commas, and terms like "/Year" were removed before conversion. Similarly, property sizes were cleaned of units like "m²" to yield integer values.

The conversion of bedroom and bathroom counts was handled using pd.to_numeric with the errors='coerce' parameter, which transformed unconvertible entries into NaN values; these missing values were then systematically replaced with "N/A" to maintain a consistent data structure for subsequent analysis.

b) Text Standardization

Location-based fields such as city, district, and province were standardized by trimming whitespace and applying consistent capitalization (e.g., INITCAP in SQL).

Redundant substrings such as “Governorate” and “District” were removed to ensure uniformity. This step was particularly important for aligning JSK and Real Estate sources, where small spelling variations (e.g., “Mount Lebanon” vs. “mount lebanon”) would otherwise fragment the data.

c) Deduplication

Although scraping scripts were designed to minimize duplicates, redundant rows inevitably appeared from refreshed or reposted property listings. To address this, deduplication rules were implemented to flag entries sharing identical values for key attributes such as address, type, size, and the number of bedrooms and bathrooms.

Finally, upon insertion into the Supabase database, an additional safeguard was provided by database-level constraints, such as the primary key on the id column, ensuring the final persistence of unique records.

d) Initial Filtering

To ensure data quality, a filtering process was applied to remove obvious outliers and invalid entries. This involved excluding listings with price or size values ≤ 0, as these were deemed erroneous, and filtering out records containing implausible attribute combinations, such as residential apartments with zero bathrooms.

Furthermore, during an exploratory cleaning phase, a small number of irrelevant listings, including those with non-Lebanese addresses (e.g., "France"), were identified and discarded.

In sum, this general cleaning process established a consistent foundation across all data sources, addressing type mismatches, text inconsistencies, duplicates, and invalid entries. These universal steps ensured that each *dataset (properties, transactions, and later synthetic data inputs)* entered the model development pipeline in a coherent, comparable state.

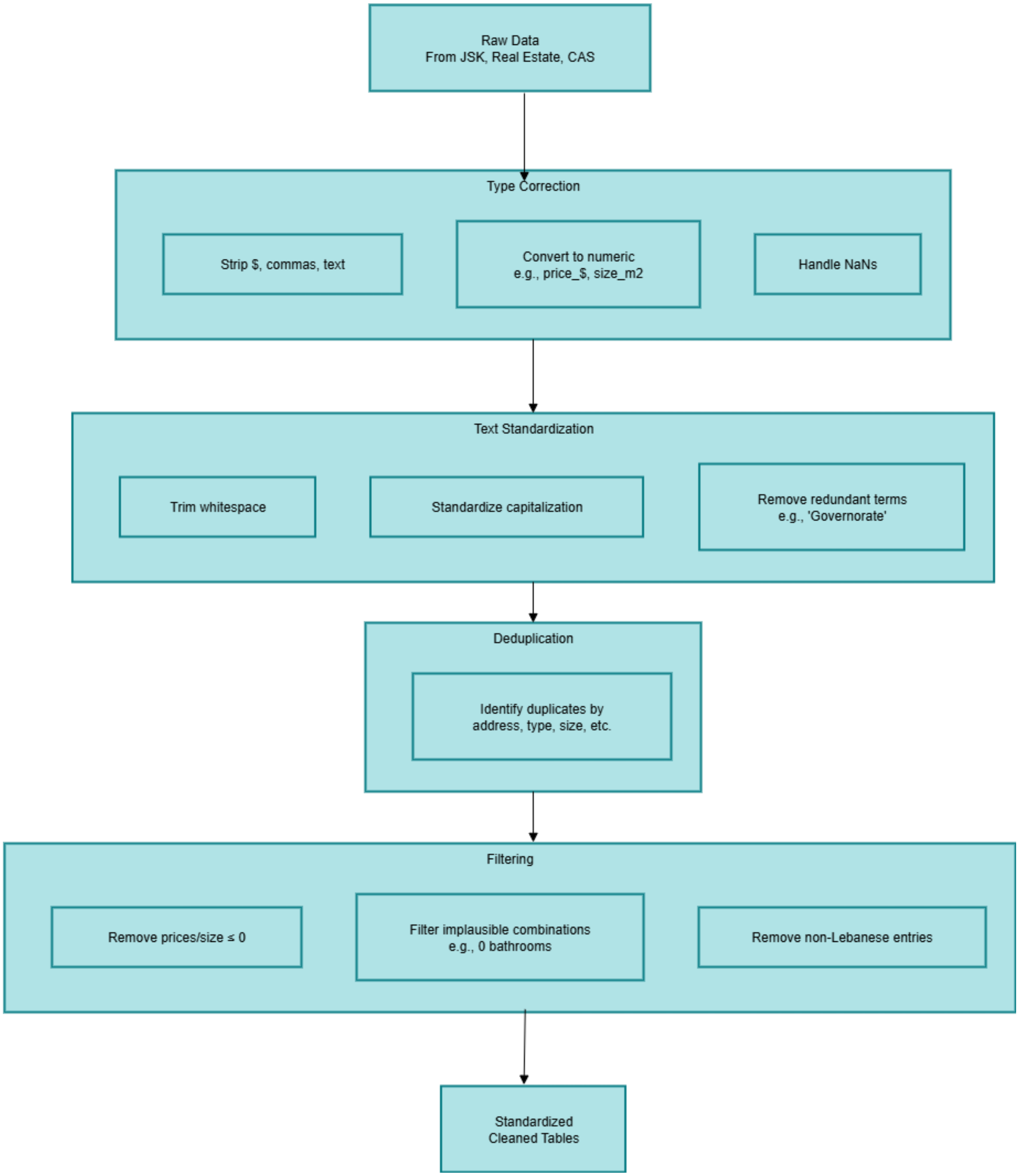


Figure 6: A high-level overview of the universal data cleaning pipeline

The cleaning applied to all raw data sources, the details of which are described in the following subsections.

3.4) Price Estimation Model

1. Dataset Context and Scope

The initial property listings dataset contained **2,849 entries** spanning multiple categories, including apartments, houses/villas, land, and various commercial assets. However, the data exhibited significant **fragmentation and regional sparsity**. For instance, houses and villas had no representation in Beirut, land listings were overwhelmingly concentrated in Mount Lebanon (88% of entries), and niche categories such as factories, gas stations, and restaurants appeared only as isolated cases.

A snapshot of the dataset composition is provided in Table 5, which highlights the disproportionate representation of property types. Apartments (n = 1,504) were by far the most common, while all other categories remained too sparse to support a generalized, multi-regional model.

Type	**Count**
Land	735
Restaurant	1
Gas Station	2
Office	129
Residential Buildings	43
Factory	6
House/ Villa	157
Chalet	116
Commercial Buildings	29
Warehouses	66
Apartments	1504
Shops	130

Table 5: Snapshot of types of real estates and their corresponding count in the dataset Properties

2. Feasibility of Modeling Different Property Types

While the initial dataset contained a variety of property categories (Table 5), not all of them were suitable for predictive modeling. An evaluation of their size, distribution, and feature relevance revealed strong disparities in feasibility.

Group 1: Relatively Feasible Candidates

- **Apartments (n = 1,504):** The largest and most consistent category, distributed across Beirut, Mount Lebanon, and North Lebanon. Apartments also share standardized attributes (e.g., bedrooms, bathrooms, size), making them the most reliable basis for predictive modeling. This category forms the empirical foundation of the price estimation model.

- **Houses/Villas (n = 157):** While smaller in scale, this category had some potential, particularly in Mount Lebanon where most listings are concentrated. However, the severe underrepresentation in Beirut (only one entry) limits the ability to generalize results nationally. A villa model could be explored as an experimental, region-specific extension, but its statistical robustness remains constrained.
- **Land (n = 735):** Land has a relatively substantial count, but its distribution is overwhelmingly biased toward Mount Lebanon (88% of entries). Moreover, land valuation depends on fewer features (primarily size and location) compared to residential units. While a simplified land model could be feasible, its geographic imbalance diminishes generalizability.

Group 2: Not Feasible with Current Data

- **Residential Buildings (n = 43) and Commercial Buildings (n = 29):** These entries contain corrupted or ambiguous attributes (e.g., "0 bathrooms") and an ill-defined target variable (total building price vs. unit-level price). Such inconsistencies prevent meaningful modeling.
- **Offices (n = 129) and Shops (n = 130):** Although sample sizes approach minimal thresholds, critical features influencing commercial valuation (e.g., foot traffic, floor level, visibility, lease terms) are absent. Models trained on only size and location would risk being highly inaccurate.
- **Warehouses (n = 66), Factories (n = 6), Gas Stations (n = 2), Chalets (n = 2), Restaurants (n = 1):** These categories suffer from extreme data scarcity. With such limited samples, no statistically sound model could be developed.

This feasibility assessment demonstrates that, despite the dataset's breadth, **apartments were the only property type with both sufficient volume and cross-regional representation to support robust and generalizable modeling**. Houses/villas and land may be considered for future, region-specific models, while all other categories were excluded due to severe sparsity, structural inconsistencies, or missing critical features.

The geographic distribution of these property types is further illustrated in Figure 7 (district-level distribution) and Figure 8 (province-level distribution). These visualizations emphasize the uneven coverage of certain categories and the relative dominance of apartment listings across regions.

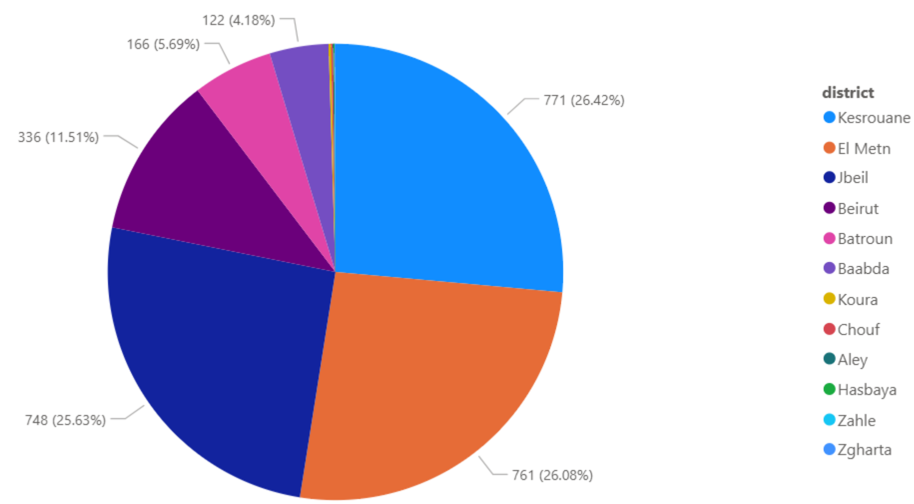


Figure 7: Distribution of All Property Types Across Districts

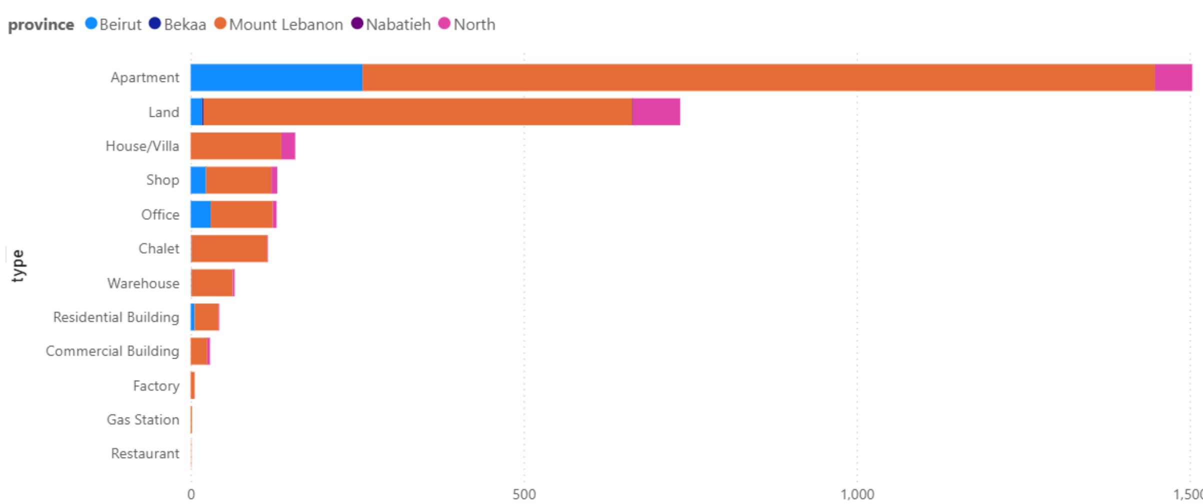


Figure 8: Stacked bar chart illustrating the distribution of property types across Lebanese provinces

Given these disparities, apartments emerged as the sole viable category for model development, with sufficient coverage across Beirut, Mount Lebanon, and North Lebanon. This strategic narrowing of scope was not a limitation of ambition but a data-driven methodological necessity to ensure statistical validity and practical utility. The preparation pipeline for the Price Estimation model was therefore designed with a deliberate focus on the apartment segment, which provided both the critical mass and geographic spread required for robust modeling.

3. Data Cleaning and Preprocessing

a. Data Filtering and Standardization

The properties dataset was first filtered to include only Apartment listings. Key numerical columns (price_$, size_m2, bedrooms, and bathrooms) were standardized through explicit type conversion. Invalid entries, such as missing price_$ or size_m2, or zero values in bedrooms or bathrooms, were removed to preserve model reliability.

b. Outlier Removal

Real estate markets often contain extreme values that can distort model learning. To mitigate this, outliers were treated specifically on the price_per_m2 feature. The 1st and 99th percentile thresholds were calculated, and all entries outside this range were excluded. This approach preserved the majority of market variation while removing the most extreme and likely erroneous records.

4. Feature Engineering

Raw property listing attributes such as size, bedrooms, bathrooms, and location provide a foundation for modeling, but they do not fully capture the structural or regional dynamics of real estate markets. To address this limitation, the apartment dataset underwent a series of feature engineering transformations designed to enrich predictive capacity, improve numerical stability, and reduce data sparsity. The process was iterative and guided by both domain logic (real estate valuation drivers) and empirical evaluation (correlation analysis).

a. Correlation Analysis

Correlation heatmaps were generated before and after feature engineering (Figures 9 & 10).

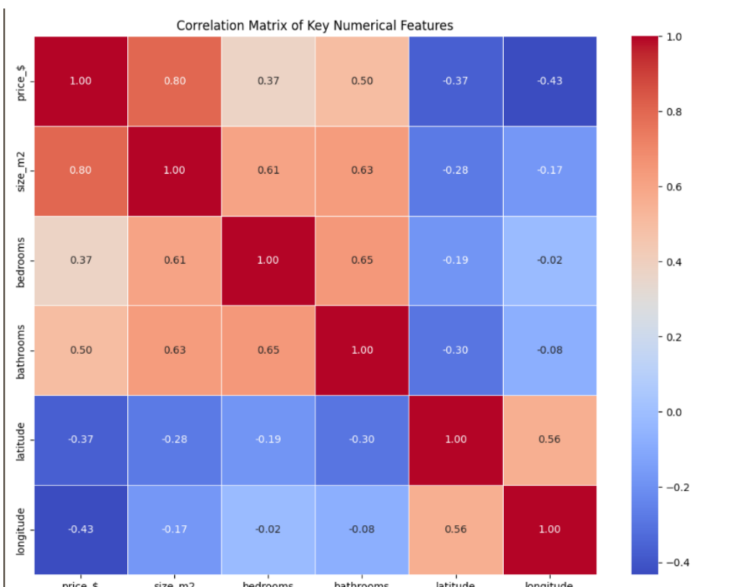


Figure 9: Correlation heatmap of raw apartment dataset

While the original dataset already showed a strong linear relationship between price_$ and size_m2 (0.80), the correlations with other key predictors like bedrooms (0.37) and raw geospatial coordinates were more moderate.

The introduction of engineered features, as shown in the second matrix, created a more robust set of predictors.

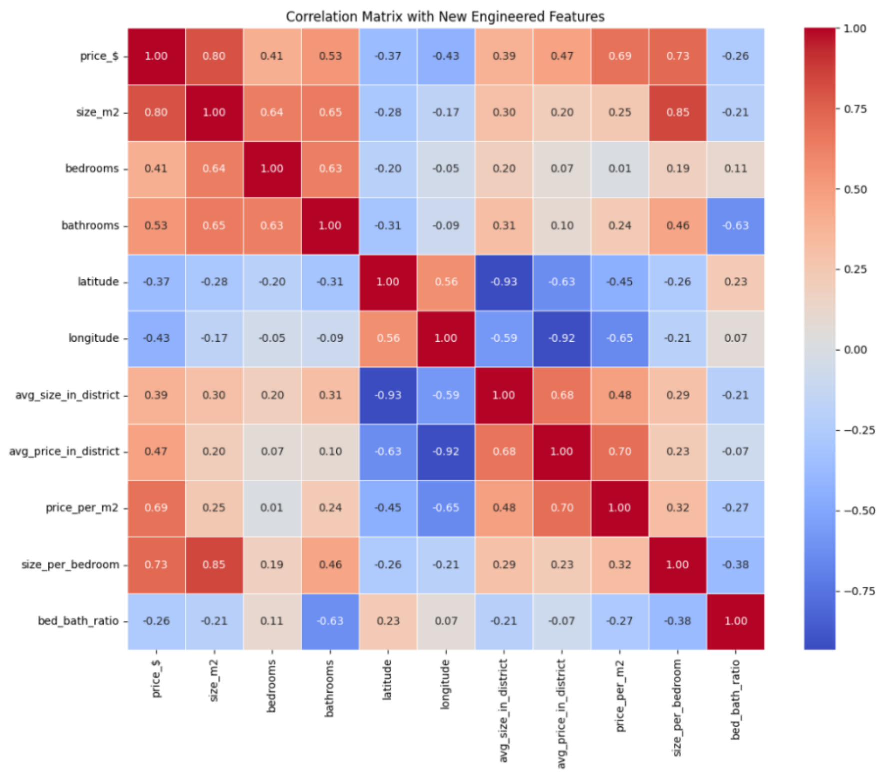


Figure 10: Correlation heatmap of engineered apartment dataset

Specifically, new features such as size_per_bedroom (0.73) and price_per_m2 (0.69) emerged as powerful indicators of price. Furthermore, the new regional aggregate feature, avg_price_in_district, effectively captured the complex influence of location, showing a much stronger relationship with latitude and longitude than the price variable itself.

These visualizations clearly illustrate the methodological benefit of feature engineering in uncovering more nuanced and predictive relationships within the data.

b. Ratio Features

Two ratio-based features were introduced to capture efficiency and spatial allocation:

- **bed_bath_ratio**: bedrooms ÷ bathrooms, representing functional balance between sleeping and utility spaces.
- **size_per_bedroom**: size_m² ÷ bedrooms, reflecting spatial efficiency per occupant.

To ensure numerical stability, division-by-zero errors and infinite values were replaced with NaN and later imputed. This avoided distortions while preserving useful variation.

c. Location-Based Aggregate Features

Real estate valuation is heavily influenced by neighborhood-level dynamics. To embed local market signals into each observation, district-level aggregates were computed using *groupby('district').transform('mean')*:

- **avg_price_in_district:** mean price per m^2 across the district.
- **avg_size_in_district:** mean property size in the district.

These variables contextualize each listing within its broader market, allowing the model to account for localized demand-supply dynamics.

d. Rare District Consolidation

Districts with < 10 apartment listings were consolidated into an **"Other"** category. This prevented overfitting to sparsely represented regions while reducing dimensionality in subsequent one-hot encoding.

e. Target and Feature Transformation

Price and size variables exhibited heavy right-skew, which violates the assumptions of many regression-based models and can bias ensemble learning. To stabilize variance and improve predictive performance, the following logarithmic transformations were applied:

- **log_price**: log(1 + price_$)
- **log_size_m2**: log(1 + size_m^2)

This transformation enabled the model to capture relative percentage changes more effectively.

f. Categorical feature Encoding

Categorical features, such as province, district, city, and property type, were encoded via one-hot encoding. The rare district consolidation (Section d) was applied before encoding to limit the number of resulting dummy variables.

g. Implementation Notes

The feature engineering pipeline was implemented in Python using pandas for data manipulation and scikit-learn's ColumnTransformer for preprocessing integration. Key steps included:

- Conversion of key columns (bedrooms, bathrooms, price_$, size_$m^2$) to numeric types, coercing invalid entries into NaN.
- Dropping of rows with missing or invalid price/size values.
- Replacement of invalid zeros in bedrooms and bathrooms with NaN.
- Outlier removal applied on price_per_m^2 (1st–99th percentile thresholds).

Through these transformations, the apartment dataset was reshaped into a **consistent, information-rich structure**. The engineered features ensured that the subsequent XGBoost model was trained not only on micro-level property attributes but also on macro-level market signals, thereby balancing granularity with contextual awareness.

3.5) Forecasting Transaction Model

The transactions dataset contained 360 rows, corresponding to 6 years of monthly observations (72 months) across 5 cities**:** Beirut, Baabda, Kesrouan, Tripoli, and Bekaa. Unlike the property listings dataset, which is heterogeneous and cross-sectional, the transactions data were temporal and sequentially structured**,** requiring a pipeline that preserved chronological dependencies while enriching the series with meaningful temporal and structural features.

1. Dataset Context and Scope

Each of Lebanon's major regions exhibits distinct rhythms in the real estate market, shaped by geography, infrastructure, migration patterns, and cultural dynamics.

- **Beirut** represents the volatile core: a high-value, low-volume market highly sensitive to political shocks and foreign investment flows.
- **Baabda** functions as a suburban magnet, offering steady mid-range growth and strong demand for apartments in proximity to Beirut.
- **Kesrouan** shows seasonal fluctuations tied to tourism and remittance cycles, with sharp spikes during summer and holiday periods.
- **Tripoli** represents a low-price, high-volume market, driven by informal housing demand and resilient to macroeconomic downturns.
- **Bekaa** remains land-driven, characterized by low liquidity and stable, agriculture-oriented valuations.

These qualitative differences manifest quantitatively in the dataset. To illustrate, Table 6 reports the average annual transaction values for each city, highlighting both cross-sectional disparities (e.g., consistently higher valuations in Beirut) and temporal variation within each region.

Year / City	2011	2012	2013	2014	2015	2016
Beirut	340.111	345.053	316.837	306.626	243.092	296.102
Baabda, Aley. Chouf	247.815	251.514	234.588	247.49	221.568	249.045
Kesrouan, Jbeil	124.092	123.847	127.241	130.472	129.648	111.902
Tripoli, Akkar	35.550	35.653	33.469	35.331	32.748	34.507
Bekaa	34.043	38.803	36.133	41.307	36.375	39.544

Table 6: Annual average transaction value by city (2011–2016)

Diagnostic plots further underscore these dynamics. For example, Figure 11 depicts Beirut's monthly transaction values over the study period, revealing jagged spikes and sharp fluctuations that motivated the design of temporal and aggregate features. By contrast, Figure 12 shows the monthly average across all cities (smoothed with a 12-month moving average), capturing the broader seasonal cycles that informed the creation of monthly_avg.

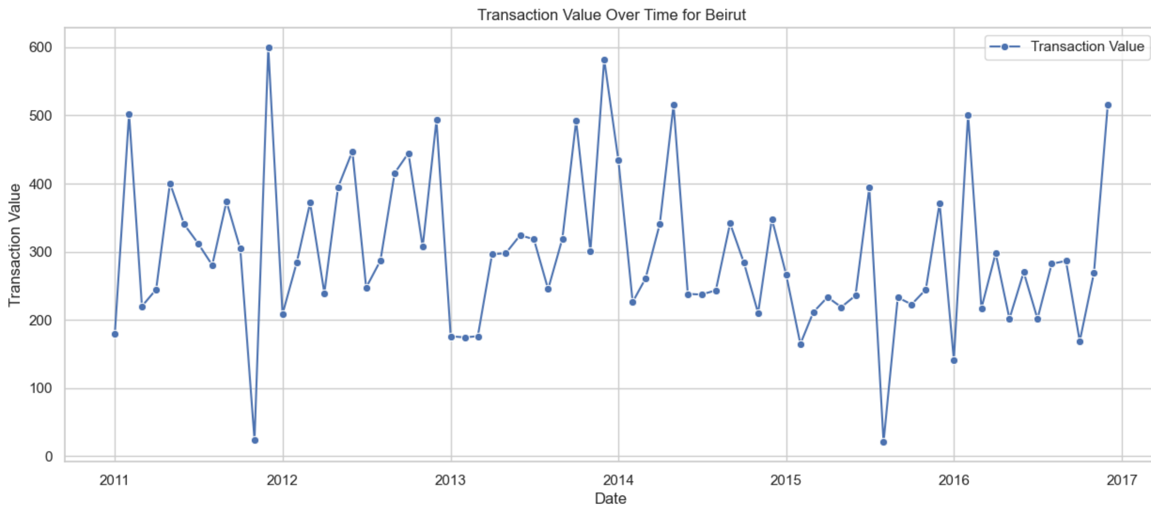


Figure 11: Raw monthly transaction values for Beirut (2011-Jan – 2016-Dec)

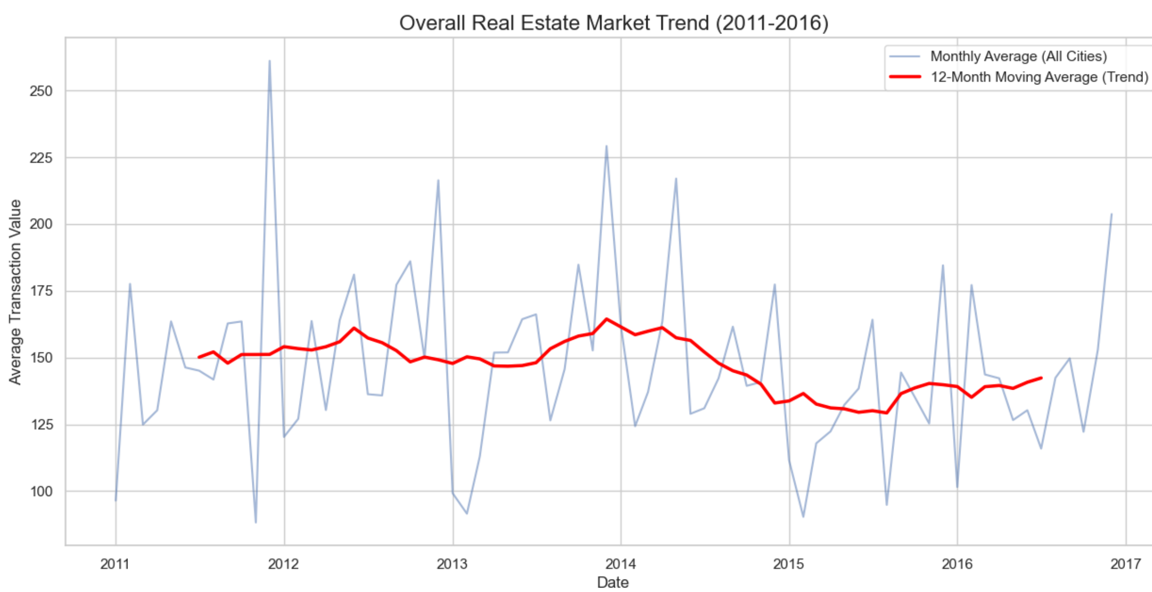


Figure 12: Smoothed monthly cross-city average transaction values

2. Data Cleaning and Preprocessing

a. Data parsing and chronology

The date column, initially stored as text (e.g., *“01-Jan-2011”*), was parsed into proper datetime objects using *pd.to_datetime*. This facilitated chronological sorting, lag-based feature construction, and seasonal decomposition. The city column was standardized to lowercase for consistency across queries and joins.

b. City name Standardization

During preprocessing, inconsistencies were identified in the naming of cities, particularly in the city column. For example, some records appeared under disaggregated or overlapping names such as *"Baabda, Alay, Chouf"* instead of the unified *"Baabda"*, or *"Tripoli, Akkar"* instead of simply *"Tripoli"*. When one-hot encoding was applied, these inconsistent names were treated as separate categories (e.g., *city_tripoli and city_akkar*), which fragmented the dataset and degraded model performance. To address this, city names were remapped to consistent, canonical labels. This ensured that each city was represented by a single feature during encoding, improving both data integrity and the robustness of the forecasting model.

3. Feature Engineering

a. Temporal Features

- time_index: sequential index capturing long-term trends.
- year, month, quarter: direct calendar features.
- month_sin, month_cos: cyclical encodings of the month variable to capture seasonality while preserving continuity between December and January.
- Binary seasonal flags: is_summer (June–August), is_december (holiday season), is_quarter_end (Q1–Q4 closings).

b. Aggregate Features

During preprocessing, inconsistencies were identified in the naming of cities, particularly in the city column. For example, some records appeared under disaggregated or overlapping names such as *"Baabda, Alay, Chouf"* instead of the unified *"Baabda"*, or *"Tripoli, Akkar"* instead of simply *"Tripoli"*. When one-hot encoding was applied, these inconsistent names were treated as separate categories (e.g., *city_tripoli* and *city_akkar*), which fragmented the dataset and degraded model performance. To address this, city names were remapped to consistent, canonical labels. This ensured that each city was represented by a single feature during encoding, improving both data integrity and the robustness of the forecasting model.

c. Structural Feature

post_2013: binary indicator for systemic market shifts following 2013, allowing the model to differentiate between pre- and post-shock dynamics.

d. Categorical Encoding

Cities were one-hot encoded after standardization, enabling the model to learn city-specific offsets and interactions.

4. Noise and Outlier Strategy

Unlike the property listings dataset, where extreme outliers could be removed, transactions are inherently volatile and often reflect meaningful shocks. Given the dataset's small size, dropping high-variance months risked discarding valuable signals. Instead, volatility was mitigated indirectly through:

- Aggregate features (monthly_avg, city_avg) which captured broader market patterns.
- XGBoost's built-in regularization , which prevents overfitting to transient spikes.

This design choice preserved the integrity of the time series while improving robustness to noise.

5. Model Choice and Implementation

XGBoost was selected as the forecasting algorithm due to its ability to:

- Handle small datasets
- Integrate heterogeneous features (temporal, categorical, structural)
- Capture non-linear dynamics

Classical time-series models (*ARIMA, SARIMA, Prophet*) and recurrent neural networks (*LSTM*) were considered, but XGBoost offered the best balance of interpretability, computational efficiency, and predictive robustness under data constraints.

The pipeline was implemented in **Python**, using **pandas** for preprocessing and **scikit-learn** for encoding and transformation. **Temporal validation** was applied to avoid leakage across time splits.

Performance metrics, forecast plots, and feature importance rankings are presented in Chapter 4 (Results).

3.6) Chat Assistant

The Chat Assistant is an interactive feature designed to democratize access to the application's analytical capabilities. Instead of requiring users to navigate complex dashboards or query structures, the assistant enables natural language interaction with the underlying models and database.

1. Design and Logic

The assistant is implemented as a React component on the frontend, connected to the Quart backend via RESTful API endpoints. User queries are parsed and routed to the appropriate analytical module (e.g., Price Estimation, CMA, Forecasting, or TimeGAN). Responses are then returned in structured JSON and rendered as conversational text.

2. Functionality

- **Natural Language Queries:** Users can ask questions such as *"What is the average price per m² in Beirut this year?"* or *"Forecast transactions in Mount Lebanon for the next quarter."*
- **Dynamic Routing:** The backend identifies the relevant model or SQL query and executes it.
- **Contextual Responses:** Results are formatted into concise, human-readable outputs, often accompanied by charts or tables.

3. Methodological Role

The Chat Assistant operationalizes the principle of **accessibility in analytics**. By abstracting away technical complexity, it ensures that insights from advanced models are usable by non-technical stakeholders, bridging the gap between AI outputs and practical decision-making.

3.7) Trend Visualization (City Circles + Price Trends)

The proliferation of digital real estate platforms has made millions of listings and transactions publicly accessible, creating unprecedented opportunities for AI-driven valuation and forecasting. However, the effectiveness of these models is often constrained by fragmented datasets, inconsistent attribute definitions, and frequent human input errors. In such contexts, visualization becomes essential: it translates noisy, complex data into interpretable signals that guide decision-making.

1. Design and Components

The **Trend Visualization module** addresses this challenge by combining two complementary layers:

- **City Circles**

  Each city is represented as a circle on a map.

  - **Size** encodes market magnitude (listing_count; i.e., no of transactions per city).
  - **Tooltips** reveal city name, trend status, and percentage change on hover.

- **City Price Trends**

  A directional arrow system compares **median prices year-over-year** to classify each city's market as:

  ✓ "Increasing"

  ✓ "Decreasing"

  ✓ "Stable"

  This allows users to track long-term growth or decline patterns and quantify percentage changes across cities.

2. Functional Role

By situating these visualizations within the broader context of fragmented and noisy real estate data, the module acts as a **bridge between raw data and strategic insight**. It enables:

- **Policymakers** to identify regional disparities
- **Investors** to spot emerging hotspots
- **Developers** to monitor long-term momentum

The design supports both **macro-level scanning** and **micro-level drill-down**, allowing users to interpret market dynamics at a glance while retaining access to city-specific detail.

3.8) Comparative Market Analysis (CMA) Tool

Alongside the AI-driven models, the application incorporates a Comparative Market Analysis (CMA) tool, deliberately designed as a traditional, non-AI component. Its purpose is to provide users with transparent, aggregated benchmarks that contextualize property values within the current market. While predictive models offer forward-looking insights, the CMA reinforces trust by grounding analysis in descriptive statistics that are both interpretable and actionable.

1. Rational and Role

The CMA tool responds to a practical need in real estate analytics: decision-makers often require market-level benchmarks as much as predictive forecasts. In volatile contexts such as Lebanon, a transparent measure of *what the market looks like today* is essential. This is not positioned as a replacement for AI, but rather as a complementary lens, a traditional baseline against which forward-looking results can be validated.

For instance, if a user intends to purchase an average apartment in Beirut, the CMA provides an immediate picture of expectations versus reality: median prices, typical sizes, and market liquidity. Such insights, presented without algorithmic abstraction, align with industry practice and facilitate trust among buyers, sellers, and analysts.

2. Implementation Logic

The CMA operates on the cleaned properties dataset stored in PostgreSQL. Its workflow is simple and transparent:

a. User Input

- Districts: two or more districts are selected for comparison.
- Property Type: limited to “Apartment,” given its statistical robustness in the dataset.
- Optional Filters: users may refine by number of bedrooms.

b. Query Execution

- The backend executes a parameterized PostgreSQL stored function (market_comparison_query).
- Filtering: properties are restricted to valid entries (price_$ >0, size_m² >0).
- Aggregation: SQL functions (e.g., PERCENTILE_CONT for medians/percentiles, COUNT(*) for activity) compute descriptive metrics.
- Grouping: results are grouped by district, returning a concise comparative dataset.

c. Output Delivery

- The processed metrics are returned to the front-end and displayed as comparative cards (for quick interpretation).
- These results are also accessible via the Chat Assistant, allowing users to request CMA comparisons through natural language queries.

3. Computed Metrics

The CMA tool generates 5 descriptive statistics that together provide a 360° view of the market:

1. **Median Price** – robust against extreme outliers, representing the typical asking price.
2. **Median Price per m²** – the standard benchmark for cross-district comparison.
3. **Typical Size** – median property size (adjusted by bedroom filter where applicable).
4. **Market Activity** – total active listings, serving as a proxy for liquidity.
5. **Price Range (10th–90th Percentile)** – captures the spread of the market, from entry-level to luxury segments.

Metric	Beirut	El Metn	Jbeil
Median Price	$650,000	$425,000	$185,000
Median Price per m^2	$2,800	$2,150	$1,250
Typical Size	195 m^2	210 m^2	160 m^2
Market Activity	450 listings	875 listings	550 listings
Price Range (10–90%)	$250k–$2.5M	$180k–$900k	$110k–$450k

Table 7: Example CMA output (apartments, three districts)

4. Workflow Summary

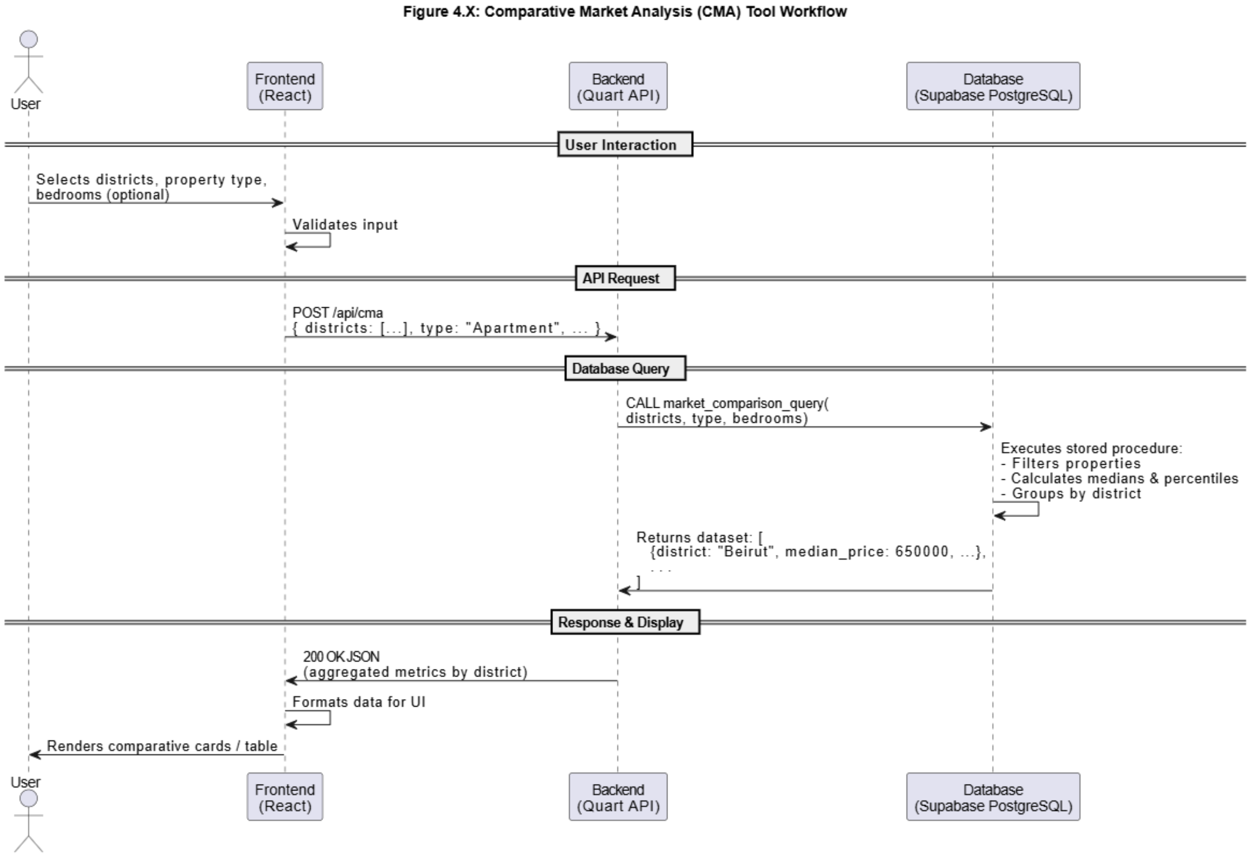


Figure 13: CMA workflow diagram

User selections trigger a parameterized query to PostgreSQL. Aggregations are computed server-side using native statistical functions. The comparative dataset is then returned to the interface for presentation as cards or tables.

5. Significance

The CMA tool represents the traditional anchor of the analytical suite. While the Price Estimator and Transaction Forecasting models project future market conditions, the CMA offers immediate, transparent insight into present-day benchmarks. This dual approach of traditional and AI-driven ensures that the platform not only provides cutting-edge predictions but also preserves interpretability and user confidence in contexts where transparency is paramount. CMA outputs also feed into the Trend Visualization module, powering City Circles and directional price arrows that reflect real-time market benchmarks.

3.9) TimeGAN: A Framework for Generating Realistic Time-series Data

1. Motivation and Core Challenge

The primary challenge in this research was the severe scarcity of historical transaction data, only 72 monthly observations (2011–2016) per city. This dataset was insufficient to train robust forecasting models, and it did not capture the unprecedented shocks of the post-2019 crisis. To extend the dataset into the 2017–2025 period in a statistically defensible way, we required a generative model capable of producing realistic, privacy-preserving, and volatility-consistent synthetic data.

The model of choice was the Time-series Generative Adversarial Network (TimeGAN). [53]

2. Rational for Selecting TimeGAN

TimeGAN was chosen over alternative generative or autoregressive models because it uniquely balances statistical realism and temporal coherence:

a. Preservation of Temporal Dynamics

Standard Generative Adversarial Networks (GANs) are highly effective at learning the distribution of static, tabular data. However, they are inherently agnostic to the sequential nature of time-series data.

They might generate data points that are individually plausible but fail to respect the crucial step-by-step conditional relationships (i.e., how a value at *time_t* depends on the value at *t-1*). TimeGAN is explicitly designed to solve this by incorporating a supervised loss that forces the model to learn these temporal dynamics.

b. Balancing Generative Flexibility and Supervised Control

The framework addresses a core dilemma in sequence modeling. Autoregressive models (like *ARIMA* or *LSTMs* used for forecasting) are excellent at learning step-by-step dynamics but are deterministic and not truly generative. Conversely, standard GANs are generative but lack explicit control over temporal correlations.

TimeGAN uniquely combines the unsupervised, generative flexibility of the GAN framework with the controlled, step-wise guidance of supervised learning.

c. Stability via Latent-Space Adversarial Training

GANs are notoriously difficult to train. TimeGAN introduces a stabilizing mechanism by conducting the adversarial game not in the high-dimensional space of the raw data, but in a lower-dimensional latent space learned by an autoencoder. This reduces the complexity of the task for the generator and discriminator, leading to more stable training and higher-fidelity outputs.

3. TimeGAN Architecture

As proposed by Yoon, Jarrett, and van der Schaar [53], TimeGAN is a composite architecture composed of 4 interconnected network components, organized into 2 functional blocks: an Autoencoder and an Adversarial Network.

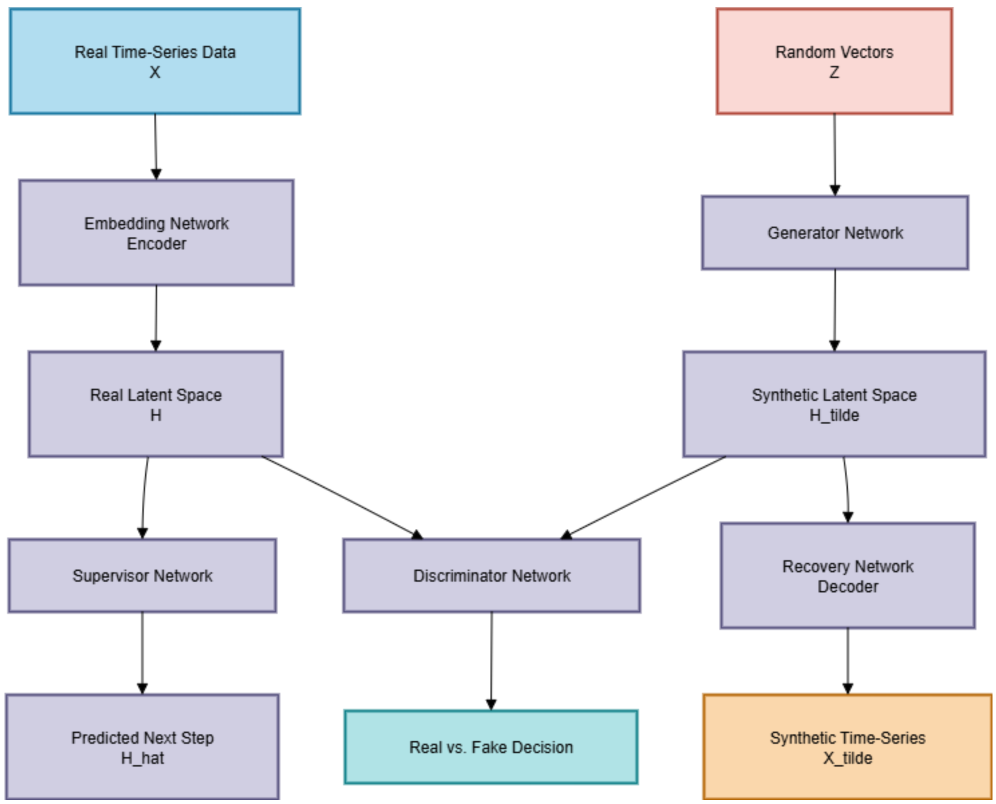


Figure 14: The TimeGAN Architecture

The TimeGAN architecture integrates an autoencoder (Embedding/Recovery) with an adversarial network (Generator/Discriminator). A Supervisor component provides an additional supervised loss to enforce temporal coherence in the latent space. [53]

a. **The Autoencoder Block:** This block learns a compressed, lower-dimensional representation of the time-series data.

Embedding Network (Encoder): Maps the real time-series data *X_tilde* into a latent space H. Its goal is to capture the essential features of the sequence in a more compact form.

Recovery Network (Decoder): Reconstructs the time-series data *X_tilde* from the latent space representation H. A successful reconstruction indicates that the latent space has effectively captured the important information from the original data.

b. The Adversarial Block: This block is responsible for generating new, synthetic data. The adversarial game is played in the latent space to improve stability.

Generator: Takes a sequence of random vectors Z as input and attempts to create a synthetic latent-space sequence *H_tilde* that is indistinguishable from the real latent sequences H produced by the Encoder.

Discriminator: Takes both real latent sequences H and synthetic latent sequences *H_tilde* as input and attempts to distinguish between them. Its feedback guides the Generator to produce increasingly realistic sequences.

4. The Joint Training Process

TimeGAN is optimized using 3 complementary loss functions:

a. Reconstruction Loss (LR):

This is a standard autoencoder loss, it ensures the autoencoder (Embedding + Recovery networks) works correctly. It measures the difference between the original data X and the data reconstructed by the Recovery network *X_tilde*. This ensures the latent space is a faithful representation of the data, and it preserves all essential information.

$$LR=E_{x\sim p(x)}[\|x-R(E(x))\|^2_2]$$

- *E*(..): Embedding network (encoder).
- *R*(…): Recovery network (decoder).
- *X:* Real data sequence.

This loss is only applied to the autoencoder components (E and R).

b. Unsupervised Adversarial Loss (LU):

This is the classic GAN loss, applied in the latent space**.** The Discriminator (D) is trained to maximize its ability to distinguish between real latent vectors (h) *(or real data)* and synthetic ones (h_tilde) *(or fake data)*, while the Generator tries to fool it.

This pushes the generated data to have the same overall statistical distribution as the real data.

i. Discriminator Loss (L_D):

The Discriminator aims to maximize this loss, correctly classifying real and generated data.

$$L_D = -Eh{\sim}p(h)[\log D(h)] - Ez{\sim}p(z)[\log(1-D(G(z)))]$$

ii. Generator Adversarial Loss(L_G):

The Generator aims to minimize this loss, fooling the Discriminator.

$$L_G = -E_{z\sim p(z)}[\log D(G(z))]$$

- h: Real latent vectors (from the Embedding network).
- *h*: Synthetic latent vectors (from the Generator),*h*= G(z).
- *z*: Random noise vector.
- *D(..)*: Discriminator's probability estimate that an input is real.
- p(h): Distribution of real latent sequences.
- P(z): Distribution of noise vectors.

c. Supervised Loss (LS):

This is the core innovation of TimeGAN.

The Supervisor network (S) learns the temporal dynamics of the real data in the latent space. It is trained to predict the next latent step given the current one. The Generator is simultaneously trained in a supervised manner. It is given a real latent sequence up to time *t-1* and is tasked with predicting the real latent vector at time t. The loss is the difference between the Generator's prediction and the actual latent vector. This directly forces the Generator to learn the step-by-step temporal dynamics of the original data, ensuring the generated sequences are not just statistically similar but also chronologically coherent.

$$LS = Eh{\sim}p(h)[\|ht+1-S(ht)\|^2{}_2]$$

- S(..): Supervisor network
- h_t: Latent vector at time t.
- ||… ||-^{2}: L2-norm (Mean Squared Error). This penalizes the Supervisor for inaccurate predictions of the next step in the sequence.

d. Total TimeGAN loss

The final objective is a weighted sum of all these components, where λ coefficients control their relative importance.

For the Generator (G), Supervisor (S), and Autoencoder (E, R):

$$\min_{G,S,E,R} L_G + \lambda_S L_S + \lambda_R L_R$$

For the Discriminator (D): $\max_D L_D$

Typical values: In the original paper, $\lambda\_S = 10$ and $\lambda\_R = 100$ are used to ensure the temporal and reconstruction fidelity are strongly enforced over the adversarial game.

5. Application in This Research: Input Pipeline & Strategic Pivots

The transactions dataset required a specialized preparation pipeline to meet the unique input requirements of the TimeGAN model, designed specifically for sequential data. This process was not only technical but also strategic, as it incorporated key pivots made to overcome early challenges such as *mode collapse* during training.

A single GAN trained jointly on all cities exhibited *mode collapse* into repetitive, nondiverse outputs. Therefore, we pivoted to a one-model-per-city design so each model could learn its city's own seasonal/scale dynamics without inter-city interference.

a. One-Model-Per-City Design

So, we had to train a **separate TimeGAN per city** (Beirut, Baabda, Kesrouan, Tripoli, Bekaa), one for each, which allowed each model to capture its city's unique "energy" (scale, volatility, seasonal rhythm).

b. Data Segregation and Input Features

The first step involved dividing the original transactions.csv dataset into city-specific subsets (e.g., beirut_data.csv, tripoli_data.csv). This ensured that each TimeGAN instance focused exclusively on the dynamics of a single market.

Each city's dataset was normalized to the range [0, 1] using a custom MinMaxScaler. Unlike standard implementations, this scaler was modified to explicitly save the minimum and maximum values used for scaling for each city (e.g., min_max_beirut.npz). This was a critical step, as it allowed the generated synthetic sequences to be accurately inverse-transformed back to their original magnitudes. This modification directly addressed a scaling bug encountered in earlier experiments, ensuring that synthetic outputs preserved their economic interpretability.

Within each file, only purely numerical features (transaction_number and transaction_value) were retained. Columns such as id and date were dropped, as TimeGAN learns temporal dependencies directly from the sequence structure rather than explicit calendar features. Similarly, the city column was removed, since the "one model per city" approach eliminated the need for within-model categorical distinctions.

The normalized two-dimensional data was reshaped into the three-dimensional tensor format required by TimeGAN: *(samples, sequence_length, features)*.

A sequence length of **24 months (2 full annual cycles)** was chosen, enabling the model to observe 2 full annual cycles. This provided sufficient context to capture seasonal patterns while avoiding overfitting to short-term noise. Using a sliding window approach, 49 overlapping training sequences were generated from the original 72 months of data ($72 - 24 + 1 = 49$), substantially augmenting the effective training set size.

c. Training Strategy for Robustness

Each city-specific TimeGAN was trained for 5,000 iterations with a batch size of 128, using GRU (Gated Recurrent Unit) cells with hidden dimensions of 24 across 3 layers. This "one model per city" strategy marked a deliberate pivot from earlier attempts to train a single, combined model, which had repeatedly collapsed due to the heterogeneous dynamics of different cities. By isolating each market, the TimeGAN models were able to learn the unique temporal signatures of their respective cities, free from cross-city distortions.

i. Summary of Architecture and Training

- Backbone: GRU-based recurrent cells (module='gru').
- Layers: 3.
- Hidden dimension: 24.
- Sequence length: 24 months.
- Iterations: 5,000.
- Batch size: 128.

TimeGAN's hybrid objective (adversarial realism + supervised next-step guidance in latent space) stabilizes training and preserves temporal coherence, which standard tabular GANs could not.

This preparation and training strategy proved successful, enabling the generation of synthetic time-series data that was both statistically realistic and visually plausible. These synthetic datasets became the foundation for the scenario simulator, offering a volatility-preserving baseline for exploring hypothetical "Boom" and "Crash" trajectories in Lebanon's volatile real estate market. It also feed into the Trend Visualization module, enabling directional comparisons and future-period City Circle updates based on simulated Boom/Crash trajectories.

*Note: Synthetic data are used **for simulation and exploration**, not as ground truth.*
We monitor for over-smoothing or drift, regenerate with fresh seeds when needed, and keep the per-city scalers/versioning so results are reproducible and auditable.

3.10) Scenario Simulator Design

The Market Scenario Simulator represents the most innovative methodological contribution of this research, purpose-built to provide explorable market narratives instead of unreliable long-range point forecasts.

The simulator generates interactive stories, unlike traditional forecasting models, which use point estimates to project the future —Reasonable Boom and Crash paths that may be interacted with by users.

In this context, traditional multi-year forecasting was deemed fragile and misleading, whereas a simulation-based approach transforms uncertainty into a structured and interactive analytical tool.

This strategic pivot arose from 2 fundamental constraints:

1. **Data Scarcity**: The available transaction dataset covered only 72 months (2011–2016), far too short and fragmented to sustain long-range forecasting.
2. **Crisis Volatility**: Lebanon's post-2019 financial collapse represented a structural break that no model trained solely on pre-crisis data could plausibly capture.

Rather than forcing fragile and misleading forecasts, the methodological decision was to transform uncertainty into an asset. The simulator reframes volatility into an educational, explorable tool, enabling stakeholders to ask: *"What if the market boomed instead of collapsing?"* or *"What if Beirut fell by 30% while Baabda remained steady?"*

This philosophy aligns with best practices in scenario-based decision-making, where simulation is recognized as a more robust strategy under high uncertainty .

The final implementation uses a Dynamic, defined-rates, Expert-Driven Simulation Model. This model applies researched, region-and-year-specific annual growth rates to a baseline volatility profile, creating realistic, nuanced, and visually compelling narratives for 5 key regions in Lebanon. These outputs are surfaced through the Chat Assistant and rendered via the Trend Visualization module, enabling users to explore Boom/Crash trajectories interactively

The idea behind this tool began with a hybrid dataset: real transaction data from 2011-2016 and synthetic data from 2017 onwards. The core challenge was that this dataset was unsuitable for traditional time-series forecasting. The real data was too limited to train a robust model, and it critically lacked the unprecedented economic shock of the 2019 crisis, making any forecast based on it fundamentally flawed.

1. Privacy and Transparency

The scenario simulator is designed with a privacy-by-design guarantee, crucially relying exclusively on synthetic data generated by TimeGAN rather than real transaction records. This approach directly addresses the rigid ethical and regulatory imperatives of handling sensitive financial information.

By design, the synthetic sequences contain no personally identifiable information (PII) or actual transaction details, effectively breaking the link between the generated data and any individual in the source dataset. This provides a formidable layer of protection against re-identification attacks, a well-documented failure of traditional anonymization techniques.

This methodology is not merely a technical workaround but aligns with cutting-edge practices advocated by leading standards bodies and industry pioneers.

Synthetic data generation offers a powerful paradigm for enabling data sharing and analysis without compromising confidentiality. Furthermore, it allows for the ethical and secure development of public-facing features, a practice endorsed by technology leaders who utilize synthetic data to innovate within strict privacy constraints. [48]

Consequently, while the simulator faithfully replicates the temporal structure, volatility, and seasonal patterns of the Lebanese real estate market, it does so without exposing a single real data point, ensuring compliance with global data protection principles and building inherent trust with end-users. This transforms a significant constraint, data confidentiality, into the core strength of a transparent and ethically sound analytical tool.

2. Theoretical Foundation: Scenario-Based "What-If" Analysis

The scenario simulator is fundamentally an implementation of scenario-based "What-If" analysis**,** a well-established methodology for decision-making under conditions of deep uncertainty. In volatile environments like the Lebanese real estate market, characterized by macroeconomic fragility and political instability, traditional point forecasting is not only unreliable but potentially misleading [29].

This tool is designed to overcome this limitation. Instead of providing a single, fragile prediction, it enables users to explore a spectrum of plausible futures based on different assumptions about macroeconomic growth or decline.

The "Boom" and "Crash" scenarios are not predictions but structured narratives, explorable answers to the critical "what-if" questions:

"What if the market grew at X% per year?" and *"What if it crashed at Y% per year?"*

This approach is aligned with strategic planning frameworks advocated by leading management institutions. As noted by McKinsey & Company, developing a range of scenarios based on critical uncertainties helps organizations "stress-test strategies and build resilience against unexpected shocks" [3].

The simulator operationalizes this principle for real estate, allowing investors, policymakers, and analysts to visually compare potential outcomes and understand the bounds of possible market behavior. Scenario outcomes are visualized using directional arrows and City Circles, allowing users to interpret regional impacts at a glance.

The value of the tool lies not in its predictive accuracy but in its ability to make uncertainty tangible and navigable**.** By interacting with the scenarios, users engage in a form of computational risk assessment, building intuition about market volatility and the potential impact of external shocks, thereby supporting more robust and informed decision-making.

3. Core Logic

At its heart, the simulator uses TimeGAN-synthesized transaction series (2017–2025) as its volatility-preserving baseline. These synthetic sequences, statistically consistent with real 2011–2016 data, form a “future history” upon which scenarios are layered (with growth/decline rates). The scenario simulator operates through a sequential, multi-stage process designed to generate plausible and volatility-preserving market narratives.

The core logic begins with the retrieval of a pre-generated synthetic baseline time-series for the user-selected city. This baseline, produced by a city-specific TimeGAN model, is statistically consistent with real transaction data from the 2011–2016 period, providing a robust and privacy-preserving foundation that recapitulate the inherent noise and seasonal patterns of the Lebanese real estate market.

Following retrieval, the simulation branches into user-defined scenarios.

The generate_scenario() function applies expert-informed annual growth or decline rates, drawn from the project's DYNAMIC_RATES dictionary, to project either a "Boom" or "Crash" trajectory. A critical step in this process is trend computation, where each annual rate is decomposed into a compound monthly growth multiplier. This multiplier is applied cumulatively to a canonical start_of_year_price to construct a smooth, overarching trend effect for the entire simulation period.

The application of the simulation is maintained through volatility preservation. Rather than replacing the synthetic baseline, the computed trend effect is additively overlayed onto it. This ensures the final output retains the month-to-month fluctuations and city-specific "jaggedness" characteristic of the original market data, preventing sterile, unrealistic projections.

The final output of this pipeline is a structured JSON object containing 3 parallel time-series: the original Baseline, the Boom scenario, and the Crash scenario.

All are subsequently rendered in comparative dashboards on the application's frontend.

This design philosophy effectively creates explorable "parallel universes": distinct yet equally plausible future trajectories, all constructed upon the same consistent synthetic history. This approach transforms inherent data uncertainty from a limitation into a structured analytical feature, allowing users to compare potential market outcomes under different conditions

4. Evolutionary Development: From Simple Curves to Narrative-Driven Simulation

The design of the scenario simulator was not conceived a priori but emerged through an iterative process of development and validation. This evolution was critical for transitioning from a simplistic, deterministic tool to a sophisticated simulator capable of generating plausible, volatility-preserving market narratives. The progression through 4 distinct versions encapsulates this refinement:

- **Version 1 (V1): Deterministic Geometric Curves.** The initial prototype implemented simple geometric growth and decline functions. This approach was rapidly abandoned as it generated sterile, smooth projections that entirely erased the market's inherent volatility, resulting in unrealistic and misleading trajectories.
- **Version 2 (V2): Volatility-Preserving Baseline.** A fundamental breakthrough was the conceptual shift to **additive scenario overlays**. Rather than replacing the synthetic baseline, this version calculated a discrete trend effect which was then added to the underlying, noisy synthetic data. This ensured the final scenarios retained the crucial "energy" and month-to-month fluctuations characteristic of the real market.
- **Version 3 (V3): Region-Aware Dynamics.** To capture the profound heterogeneity between Lebanese regions, the simulator was enhanced to apply city-specific annual growth rates. This acknowledged that a market shock in Beirut would manifest differently than in Tripoli or Bekaa, allowing the tool to model divergent regional economic dynamics.
- **Version 4 (V4): Dynamic, Time-Variant, Expert-Driven Model (Final).** The current and final version integrates these lessons into a comprehensive framework. It

introduces a DYNAMIC_RATES dictionary that maps specific annual growth rates to each city and each year, enabling the simulation of complex, multi-phase narratives like Lebanon's post-2019 economic collapse and tentative recovery. To further enhance realism, a minor random jitter (±1.5%) is applied to each monthly value, ensuring naturalistic, non-parallel scenario trajectories and preventing an overly smooth, artificial appearance.

This iterative development process was essential for creating a tool that moves beyond naive projection and serves as a powerful medium for exploring data-driven, plausible market narratives.

Year	Beirut	Bekaa	Tripoli	Baabda	Kesrouan	Validation Source(s)
2019	Crash: −7% Boom: +3%	Crash: −5% Boom: +2%	Crash: −6% Boom: +3%	Crash: −4% Boom: +3%	Crash: −3% Boom: +4%	Reuters on early financial collapse ; Carnegie on policy failures. [60] [61]
2020	Crash: −17.5% Boom: +5%	Crash: −12.5% Boom: +4%	Crash: −15% Boom: +5%	Crash: −12% Boom: +6%	Crash: −10% Boom: +7%	World Bank: "Lebanon Sinking to the Top 3" ; Reuters on sovereign default [59][60]
2021	Crash: −10% Boom: +7.5%	Crash: −8% Boom: +6.5%	Crash: −9% Boom: +7.5%	Crash: −7% Boom: +8.5%	Crash: −6% Boom: +9.5%	L'Orient Today on real estate paradox ; National News on regional disparities [62] [63]
2022	Crash: −3.5% Boom: +10%	Crash: −2.5% Boom: +8.5%	Crash: −3.5% Boom: +9.5%	Crash: −2% Boom: +11%	Crash: −2% Boom: +12%	IMF/World Bank on partial stabilization ; HBR on simulation for risk. [59][64]
2023	Crash: −1% Boom: +12.5%	Crash: −0.5% Boom: +10%	Crash: −1% Boom: +11%	Crash: −0.5% Boom: +12%	Crash: +2% Boom: +14%	L'Orient Today on "safe haven" behavior ; Investopedia on what-if analysis. [62][65]
2024	Crash: +2% Boom: +15%	Crash: +3% Boom: +12.5%	Crash: +2% Boom: +13.5%	Crash: +3% Boom: +14.5%	Crash: +4% Boom: +17%	World Bank on fragile recovery ; National News on suburban resilience. [59][63]
2025	Crash: +3% Boom: +17%	Crash: +4% Boom: +15%	Crash: +3% Boom: +16%	Crash: +4% Boom: +17%	Crash: +5% Boom: +18.5%	Carnegie & IMF forward-looking assessments ; HBR on scenario planning. [61][64]

Table 8: Annual Growth/Decline Rates Used in the Scenario Simulator (2019–2025)

Note: Rates are expressed as annualized percentage changes applied per city. Negative values represent Crash scenarios, while positive values represent Boom scenarios.

The simulator does not learn these scenarios from data. Instead, it applies expert-informed annual rates, ensuring interpretability and narrative fidelity. For example, the post-2019 period can be depicted as a sharp collapse followed by stabilization and partial recovery, directly aligning with known economic events.

Chapter 4: Results and Findings

This chapter presents the empirical results derived from the methodologies described in Chapter 3. Its purpose is to directly address the research questions posed in the introduction by systematically evaluating the outputs of the developed models and tools. The chapter is organized into three parts.

First, the performance of the predictive models is examined, beginning with baseline ensemble experiments and culminating in the optimized XGBoost estimators for both property prices and transaction forecasting.

Second, the adequacy of synthetic data generation using TimeGAN is assessed. Both quantitative metrics and qualitative visualizations followed by an exploration of the Market Scenario Simulator's outputs.

Finally, the integrated web application is showcased, illustrating how the individual analytical components were deployed into a coherent, user-facing system.

4.1) Performance of Predictive Models

4.1.1. Price Estimation Model Results

The initial approach to real estate price estimation involved training a generalized ensemble model across all available property types to establish a baseline performance. This preliminary experiment quickly highlighted the severe limitations of the raw, heterogeneous dataset, particularly the extreme class imbalance and sparsity for many property categories.

As presented in Table 9, the initial ensemble model yielded overall poor predictive performance across most property types. While apartments showed a moderate R^2 of 0.4383, indicating some predictive capability, other categories like "House/Villa" and "Land" had very low R^2 values (0.0364 and 0.2838, respectively), and "Residential Building" even resulted in a negative R^2 of -0.0072, signifying that the model performed worse than simply predicting the mean. These results provided crucial empirical evidence necessitating a strategic narrowing of the project's scope to focus on the "Apartment" property type, which possessed the most viable data for robust modeling.

Type	R2	RMSE
Apartments	0.4383	196457.44
Chalet	0.2461	230626.76
Commercial Buildings	0.2613	458029.4
House/Villa	0.0364	961194.31
Land	0.2838	2548325.58
Residential Building	-0.0072	836613.34

Table 9: Initial Ensemble Model Performance (All Property Types)

Following the decision to focus solely on apartment price estimation, a comparative analysis was conducted between 2 leading tree-based ensemble models: LightGBM and XGBoost. Both models were trained on the meticulously preprocessed and feature-engineered apartment dataset (as detailed in Section 3.3.2) to identify the superior performer.

Table 10 summarizes the results of this bake-off. XGBoost consistently outperformed LightGBM across all key metrics, achieving a higher best cross-validated R^2 score of 0.8064 compared to LightGBM's 0.7739. On the test set, XGBoost further demonstrated its superiority with an R^2 of 0.7879, a notable improvement over LightGBM's 0.7588. Additionally, XGBoost yielded a lower Mean Absolute Error (MAE) of $103,732.10, indicating more accurate predictions on average than LightGBM's $113,613.79. Visualizations of feature importance (Figure 15 and Figure 16) and regression plots (Figure 17 and Figure 18) further supported the selection of XGBoost by illustrating its robust feature learning and closer alignment of predicted versus actual values.

Metrics	LightGBM	XGBoost
Best R2 score on cv	0.7739	0.8064
R2	0.7588	0.7879
MAE	113 613.79	103 732.10

Table 10: Comparative Performance of LightGBM vs. XGBoost for Apartments

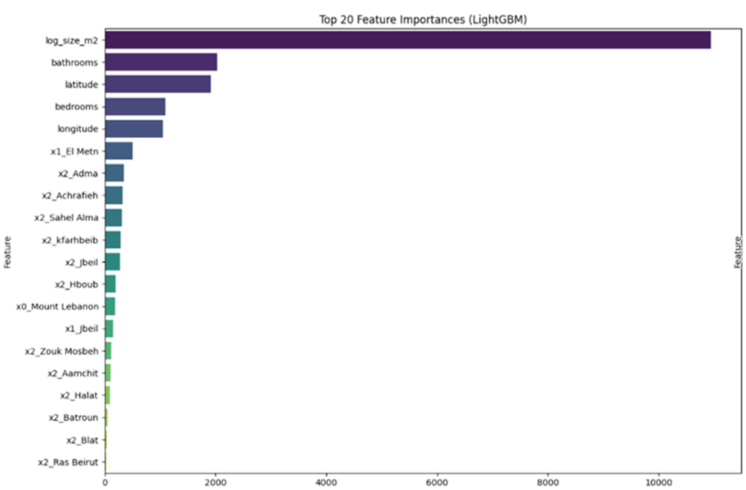


Figure 16: Top 20 feature importance by XGBoost model

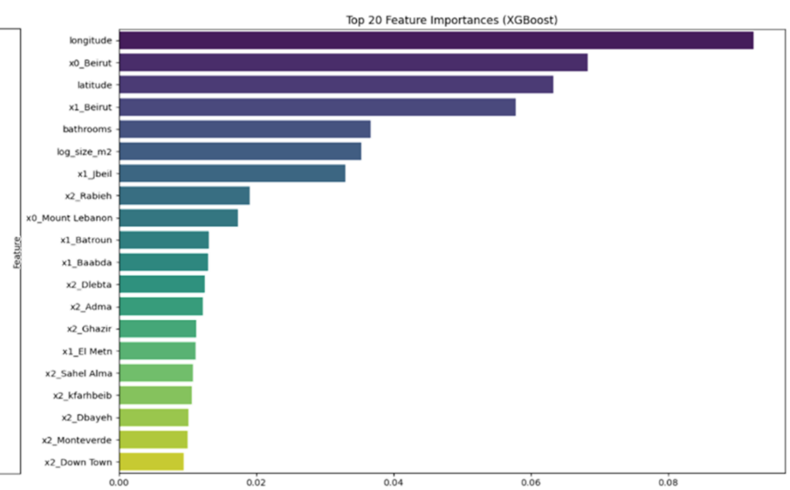


Figure 15: Top 20 feature importance by LightGBM model

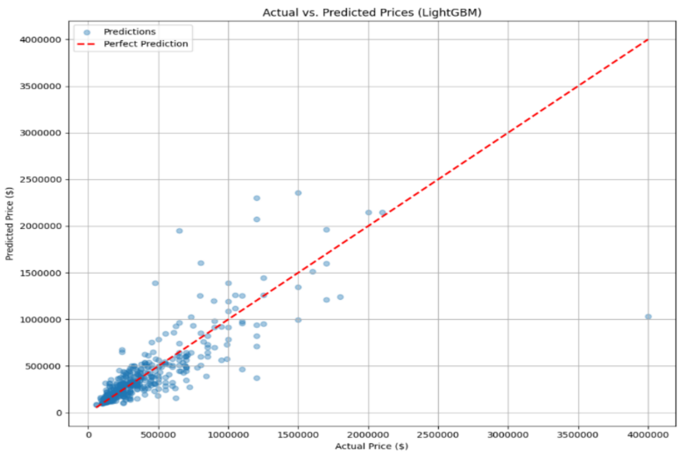


Figure 17: Regression plot by LightGBM model

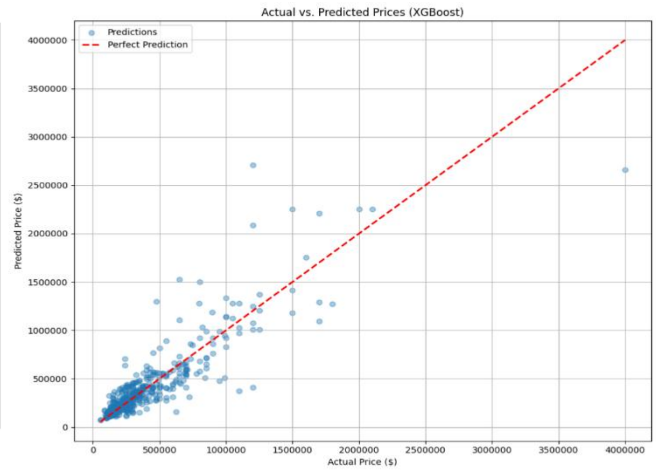


Figure 18: Regression plot by XGBoost model

The final optimized XGBoost model for apartment price estimation, refined through extensive hyperparameter tuning (as detailed in Section 3.4.1), demonstrated strong predictive capabilities. The final evaluation metrics for this model were: a best cross-validated R^2 of 0.7937, an R^2 on the test set of 0.7733, and a Mean Absolute Error (MAE) of $84,896.95. These metrics collectively indicate that the model effectively captures a significant portion of the variance in apartment prices and provides predictions with a reasonable margin of error, especially within the volatile Lebanese market context.

The most influential features driving price predictions, as identified by the model, are presented in Figure 19, offering insights into market dynamics. Furthermore, the relationship between the model's predicted prices and the actual prices for the test dataset is visually represented in a scatter plot (Figure 20), where points are observed to cluster tightly around the diagonal line, confirming a strong positive correlation and reliable predictive performance.

Metrics	XGBoost
Best R2 score on cv	0.7937
R2	0.7733
MAE	$84,896.95

Table 11: Final XGBoost Model Performance for Apartments

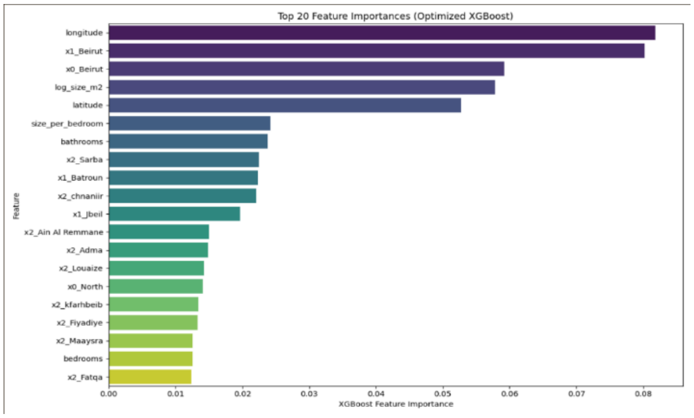


Figure 19: Top 20 feature importance by final XGBoost model

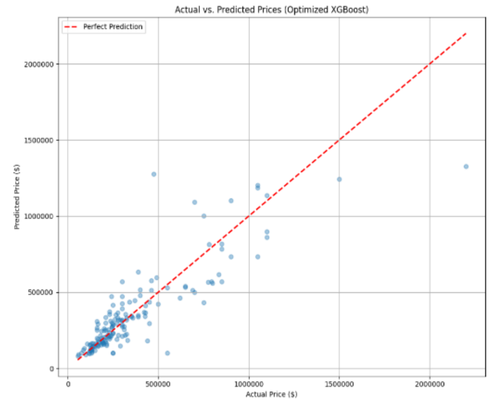


Figure 20: Regression plot by the final XGBoost model l

Beyond tabular metrics and regression plots, the apartment price estimation model is also deployed through the Chat Assistant, enabling natural language interaction with the predictive engine. Users can query the system conversationally, for example: *“Estimate the fair price of a 120 m² apartment in Achrafieh with 2 bedrooms and 2 bathrooms.”* The assistant processes the request, routes it to the optimized XGBoost estimator, and returns the predicted price along with contextual benchmarks such as median price per m² in the selected district.

This conversational interface lowers the barrier to entry for non-technical users, who may not wish to navigate dashboards or interpret scatter plots. It also enhances interpretability by embedding the model’s outputs within a dialogue that can clarify assumptions, explain feature drivers, or compare results across districts. In practice, this means that the same robust predictive model evaluated above is not only technically validated but also operationalized in a user-friendly, accessible format that supports decision-making in real time.

4.1.2. Transaction Forecasting Model Results

The transaction forecasting task presented unique challenges due to the small and noisy nature of the transactions.csv dataset. Initially, the potential of Long Short-Term Memory (LSTM) networks was considered, given their theoretical power in sequence modeling. However, empirical evaluation quickly revealed that for this specific dataset, a simpler yet robust model, XGBoost, significantly outperformed LSTM. As shown in Table 12, XGBoost achieved a notably lower RMSE (58.90 vs. 176.10) and MAE (30.35 vs. 127.01) compared to LSTM, demonstrating its superior ability to extract meaningful patterns without overfitting to the noise in the limited historical data. This comparative analysis provided strong justification for proceeding with XGBoost as the primary forecasting algorithm.

Metrics	LSTM	XGBoost
RMSE	176.10	58.90
MAE	127.01	30.35

Table 12: Comparative Performance of LSTM vs. XGBoost for transaction forecasting

An initial XGBoost model, trained on the pre-2016 data and evaluated on the 2016 test year, yielded a Mean Absolute Percentage Error (MAPE) of 18%.

Following this, the model was further optimized by incorporating a refined set of temporal and aggregate features (as detailed in Section 3.3.3), designed to capture the underlying seasonal and regional patterns more effectively without relying on potentially disruptive lag or rolling features. This refined XGBoost model demonstrated robust performance on the validation set, as summarized in Table 13.

Metrics	XGBoost
RMSE	58.90
MAE	30.35
MAPE	22%

Table 13: Final XGBoost Model Performance for Transaction Forecasting (2016 Test Year)

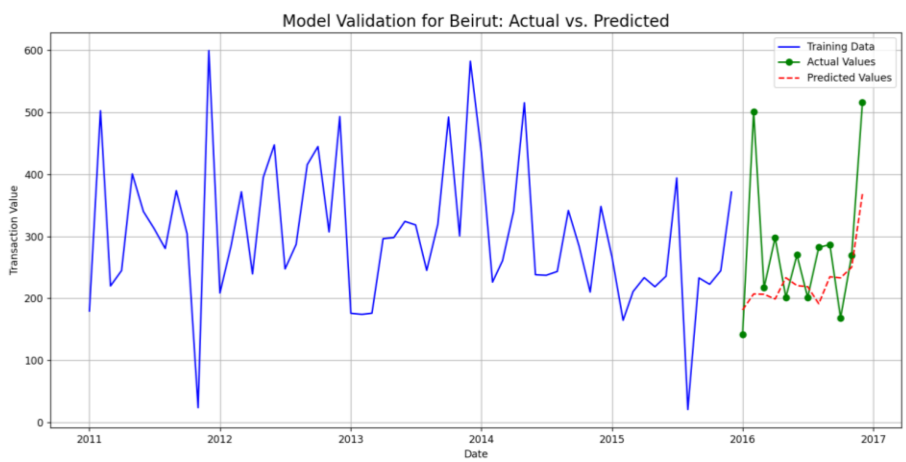


Figure 21: Model Validation for Beirut: Actual vs. Predicted by initial XGBoost model (2016 test year)

Figure 21 visually presents the model's initial validation for the Beirut region in 2016, illustrating its ability to capture the general seasonal peaks and troughs. After comprehensive feature engineering and tuning, Figure 22 showcases the enhanced predictive performance for the final updated XGBoost model for Beirut, with the model successfully capturing the seasonal trends and overall trajectory of transaction values over the 2016 test period.

The most influential features in driving these transaction forecasts, as identified by the optimized XGBoost model, are presented in Figure 23, offering insights into the market dynamics the model learned.

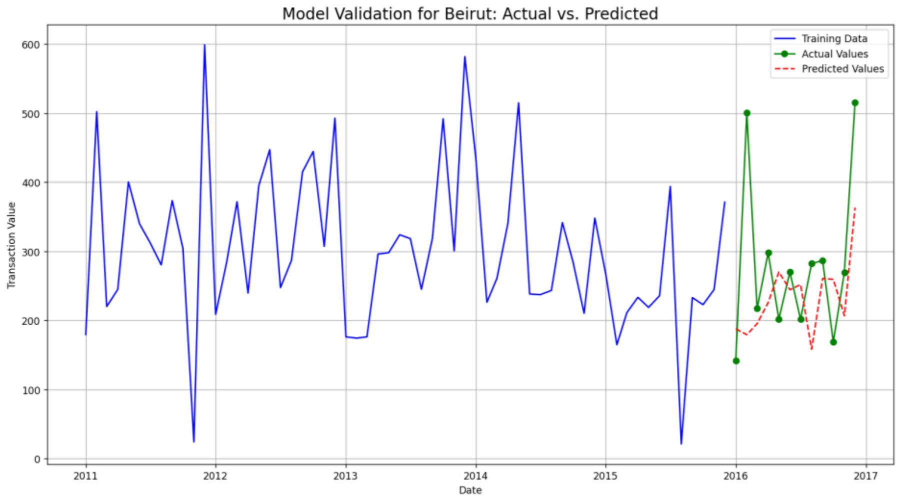


Figure 22: Model Validation for Beirut: Actual vs. Predicted by final updated XGBoost model (2016 test year)

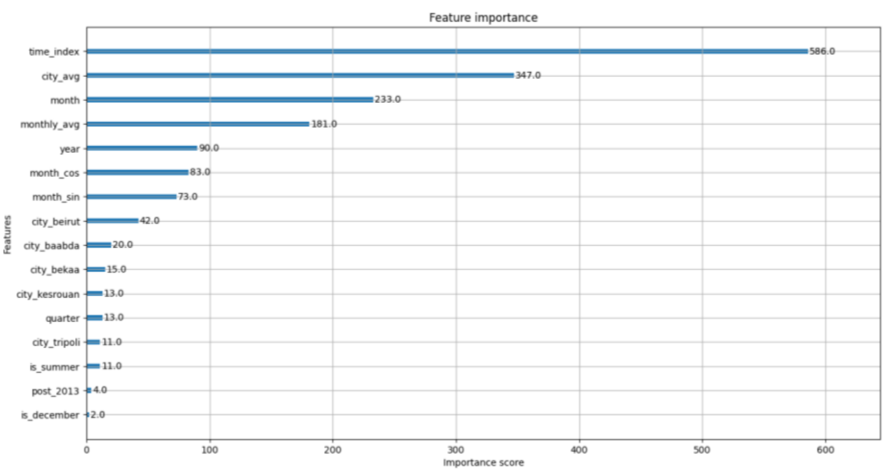


Figure 23: Top feature importance by the final updated XGBoost model

The transaction forecasting model is also accessible through the Chat Assistant, enabling users to request forecasts in natural language. For example, a user might ask: “How many property transactions are expected in Beirut next year?” The assistant interprets the query, routes it to the XGBoost forecasting engine, and returns the projected transaction count along with contextual insights such as seasonal trends or historical comparisons.

This conversational interface makes forecasting outputs accessible to non-technical users and supports real-time exploration of market dynamics. By embedding model results into a dialogue, the assistant enhances interpretability and allows users to probe specific regions or timeframes without navigating complex dashboards.

4.2) Evaluation Of The Synthetic Data Solution

This section provides the empirical evidence for the strategic necessity of pivoting from direct long-range prediction to simulation (addressing RQ2) and evaluates the effectiveness of the TimeGAN-based approach in generating realistic synthetic data (addressing RQ3).

4.2.1. Inadequacy of Real Data For Simulation

To understand why direct, robust long-range scenario simulation was infeasible with the raw data, a visual analysis of the historical transactions.csv dataset across different regions is crucial.

As shown in Figure 11 (e.g., Beirut from 2011 to 2016) , the transaction data for Beirut, a high-value and highly volatile market, is characterized by significant month-to-month noise and covers a limited time frame, with its extreme variability reflecting high sensitivity to political shocks and foreign investment flows.

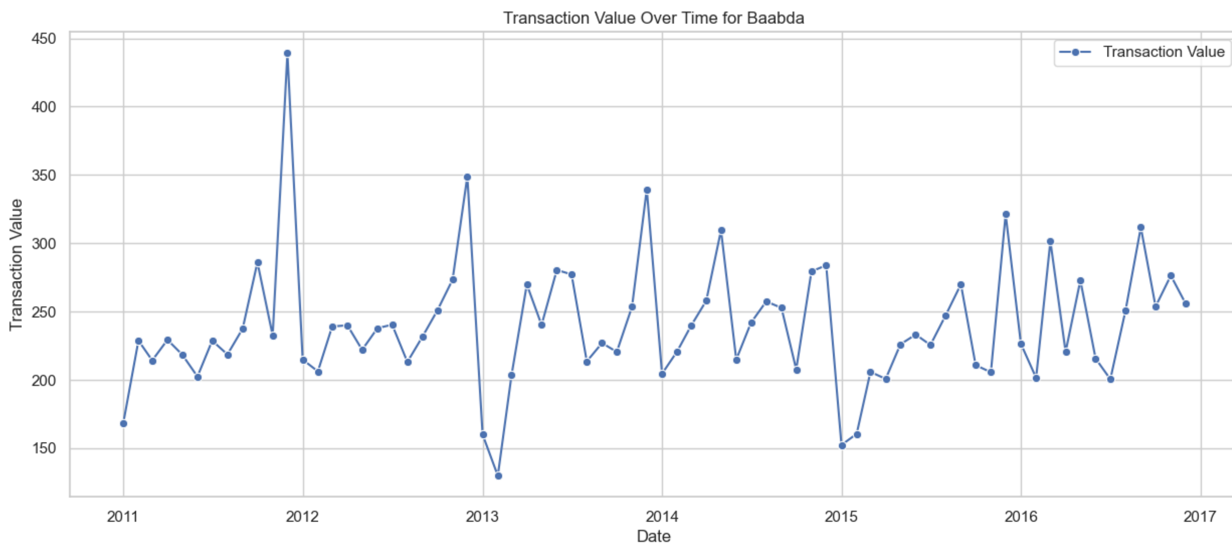


Figure 24: Raw monthly transaction data for Baabda (2011-2016)

As shown in Figure 24, the transaction data for Baabda, showing mid-range activity with notable volatility, is characterized by significant month-to-month noise and covers a limited time frame, with its fluctuations demonstrating responsiveness to broader national confidence and local demand.

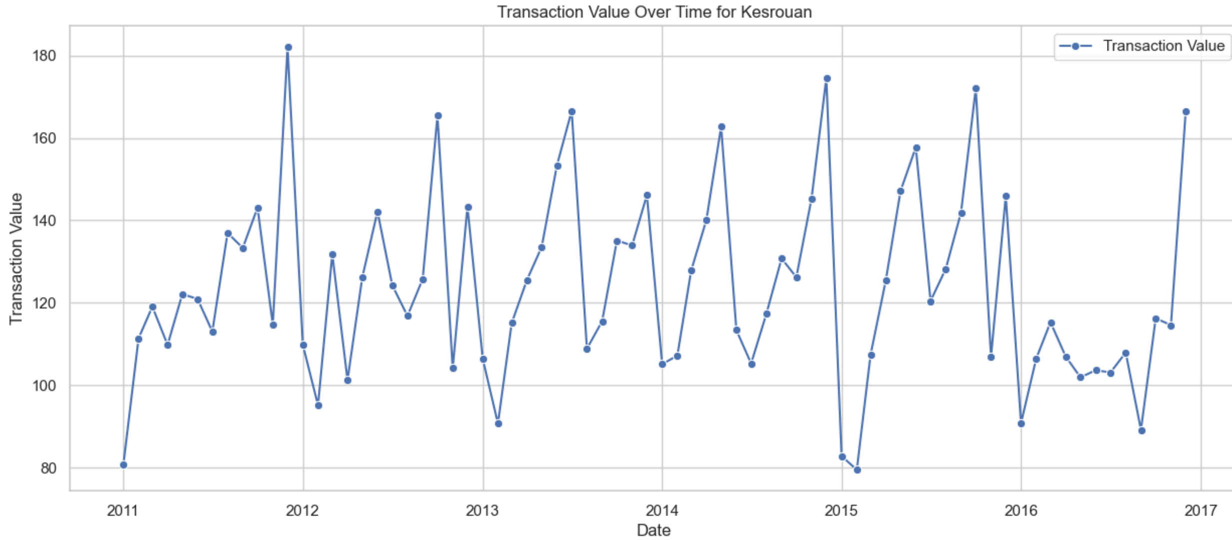


Figure 25: Raw monthly transaction data for Kesrouan (2011-2016)

As shown in Figure 25, the transaction data for Kesrouan, exhibiting moderate values with sharp, often cyclical, peaks and troughs, is characterized by significant month-to-month noise and covers a limited time frame, with its variability strongly tied to tourism and remittance cycles.

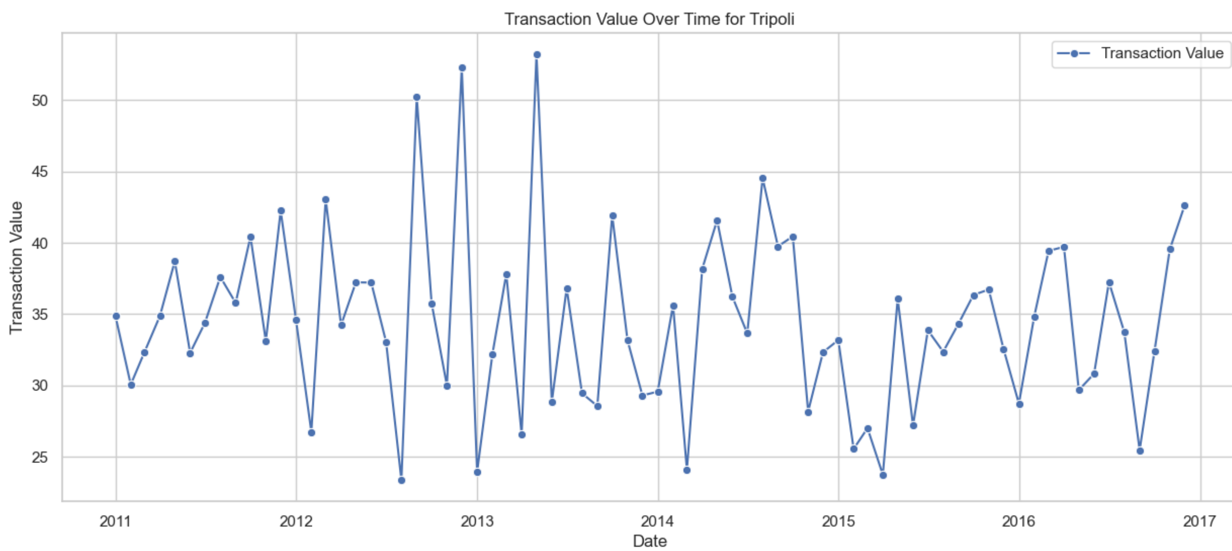


Figure 26: Raw monthly transaction data for Tripoli (2011-2016)

As shown in figure 26, the transaction data for Tripoli, consistently low in absolute value but highly active, is characterized by significant month-to-month noise and covers a limited time frame, with its pronounced variability showcasing resilience amidst persistent local and regional challenges.

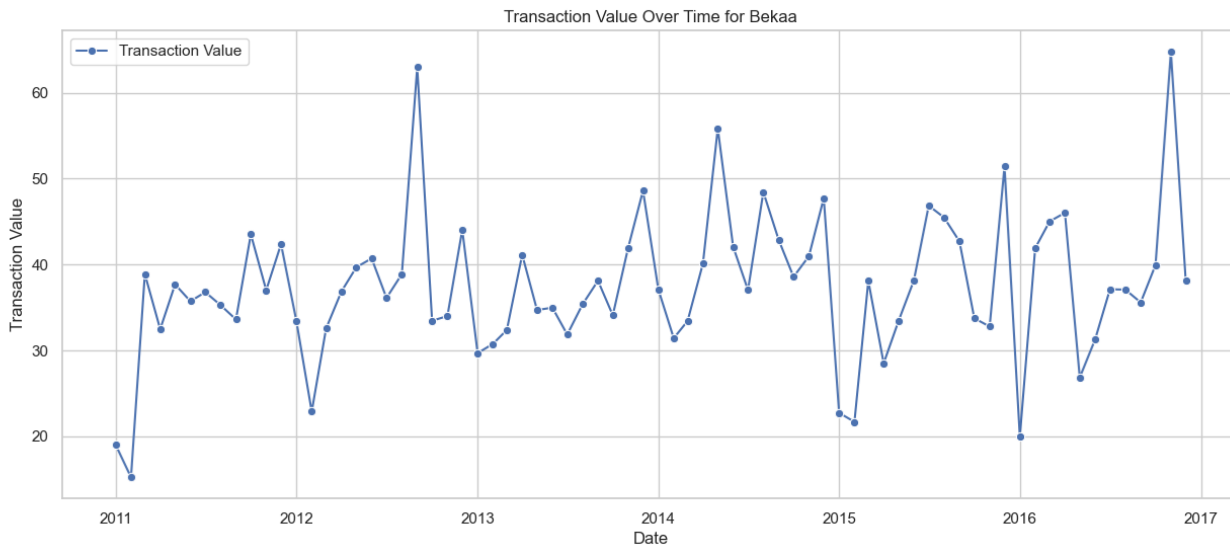


Figure 27: Raw monthly transaction data for Bekaa (2011-2016)

As shown in Figure 27, the transaction data for Tripoli, consistently low in absolute value but highly active, is characterized by significant month-to-month noise and covers a limited time frame, with its pronounced variability showcasing resilience amidst persistent local and regional challenges.

This scarcity of reliable historical context, particularly the absence of data preceding and during the unprecedented post-2019 economic crisis in Lebanon, renders this dataset statistically unsound for training traditional predictive models capable of generating robust, long-range market scenarios. Any attempt at direct long-term forecasting or simulation from this raw data would be highly speculative and prone to extreme inaccuracies.

4.2.2. TimeGAN Output Analysis

The implementation of TimeGAN was rigorously evaluated to ensure its effectiveness in generating realistic, synthetic time-series data that preserves the statistical properties and temporal dynamics of the original data. This evaluation directly addresses RQ3, demonstrating how the chosen "one model per city" strategy successfully overcame the challenges encountered in initial attempts.

Early experiments involved training a single TimeGAN model across all cities. This approach, however, suffered from severe mode collapse, where the synthetic data failed to capture the full diversity and distinct temporal characteristics of each city. Visualizations, such as t-SNE projection (Figure 28), clearly showed synthetic data points clumping into dense, repetitive structures, with little overlap or interleaving with the widely dispersed real data.

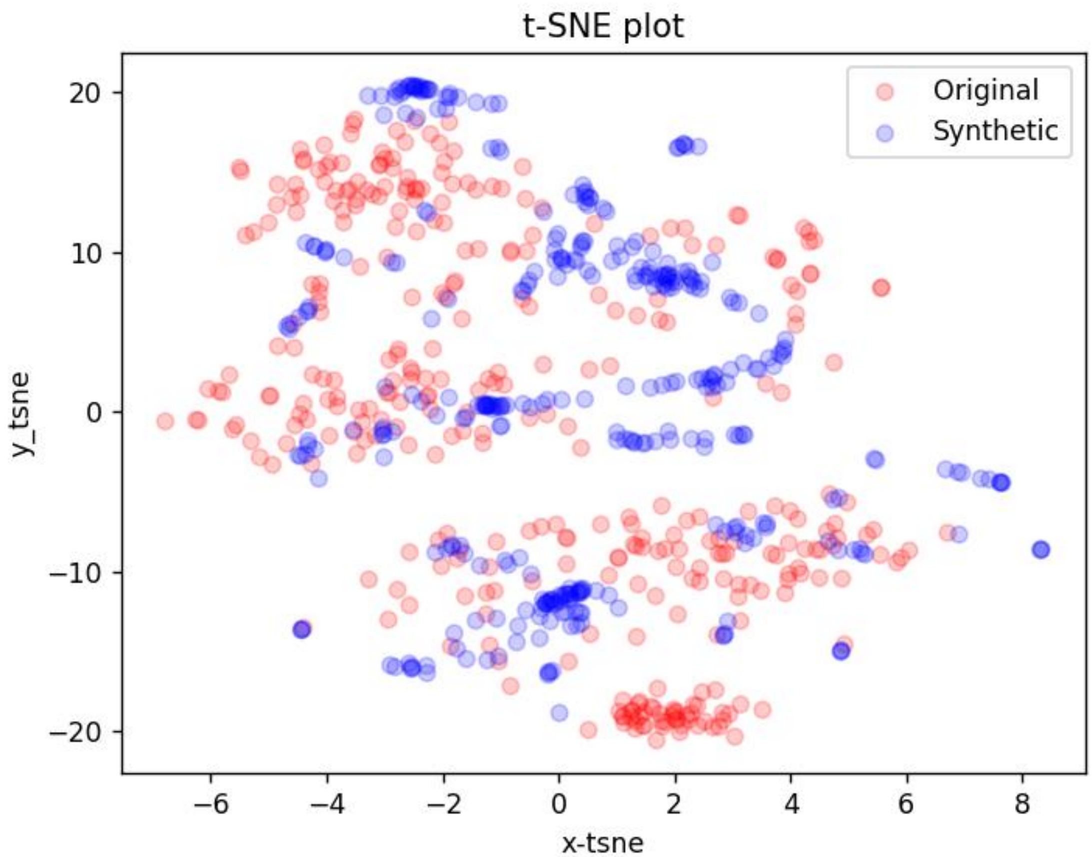


Figure 28: t-SNE projection of real vs. synthetic data (failed single-model TimeGAN, merged cities)

This qualitative evidence of mode collapse was quantitatively supported by low discriminative scores, indicating a clear distinction between real and synthetic sequences by a simple classifier.

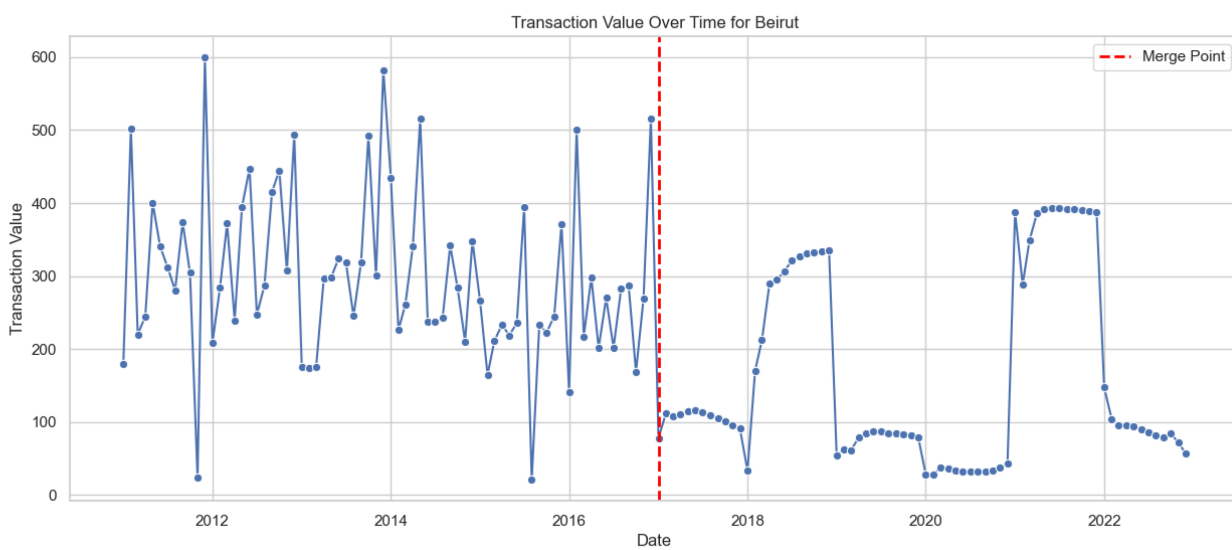


Figure 29: Beirut Transaction Value: Original Data vs. Multi-City TimeGAN Synthes

Figure 29 displays the original transaction values for Beirut from 2011 up to approximately early 2017 (before the red "Merge Point"), alongside synthetic transaction data generated for Beirut by a TimeGAN model trained on a combined dataset of all cities, from early 2017 to 2022.

The plot shown in figure 29 provides clear visual evidence that the "one model for all cities" approach for TimeGAN leads to a mismatch in replicating the distinctive characteristics of individual cities. For Beirut, the synthetic data generated lacks the high-frequency, high-amplitude volatility and extreme peak values that are defining features of its original transaction history, indicating that the model failed to preserve Beirut's unique economic signature when trained on a heterogeneous dataset.

In contrast, the refined "one model per city" TimeGAN approach (detailed in Section "*feature engineering of price estimation models*") proved highly effective. Figure 30 presents the t-SNE projection for a successful per-city TimeGAN model (e.g., for Beirut), where the synthetic data points (blue) are highly interleaved and intermingled with the original data points (red).

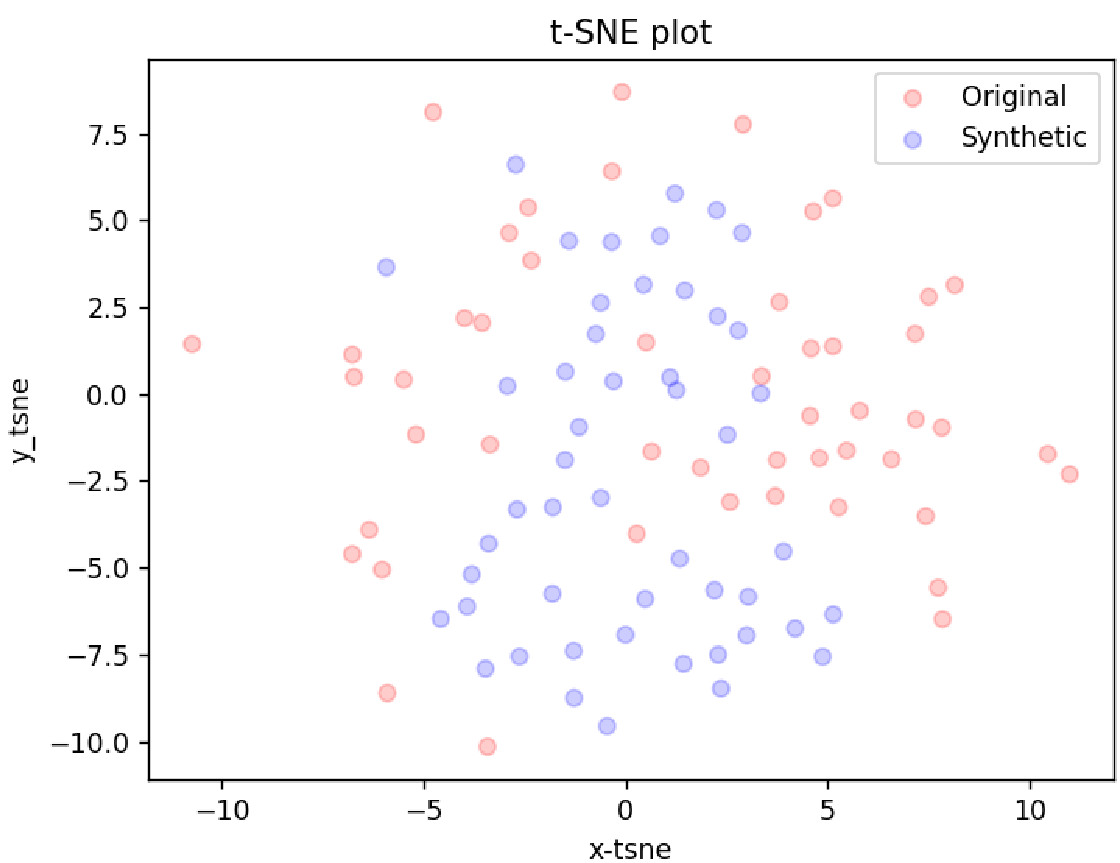


Figure 30: t-SNE projection of real vs. synthetic data (successful per-city TimeGAN, e.g., Beirut)

Quick analysis: Unlike the multi-city t-SNE plot which showed potentially distinct clusters for different cities, this single-city plot presents a more homogeneous, well-blended distribution, which is ideal when the goal is to replicate the characteristics of a single, unified dataset.

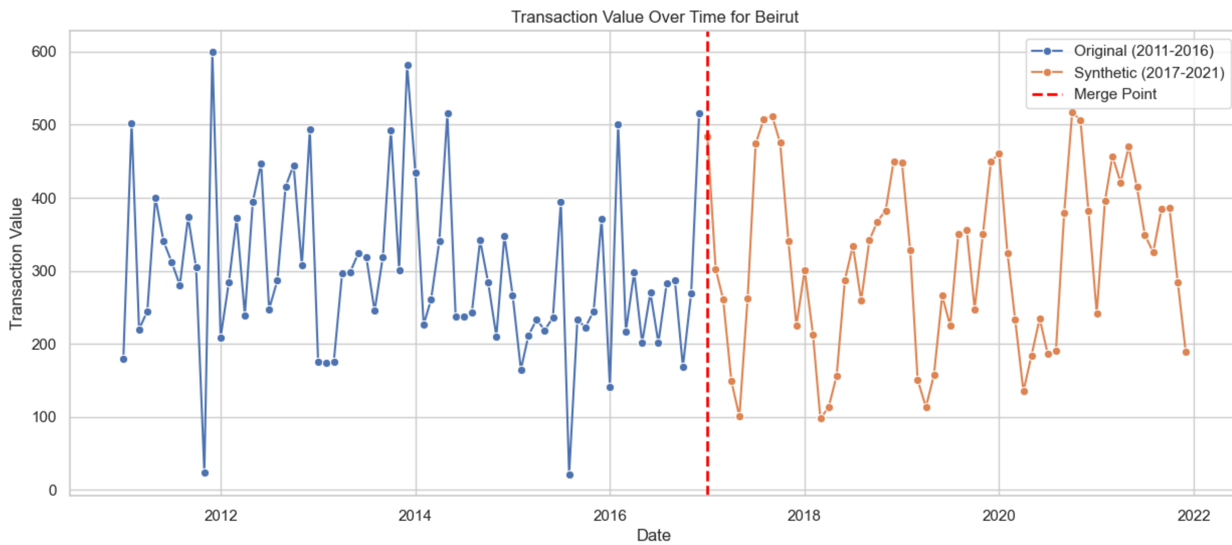


Figure 31: Beirut Transaction Value: Original Data (2011-2016) vs. Single-City TimeGAN Synthetic Data (2017-2021)

The plot shown in figure 31 tells us that the synthetic data effectively captures Beirut's unique economic signature: its extreme volatility, high transaction values, and rapid shifts.

These visuals evidence confirms that the generative model successfully learned the underlying distribution and temporal dynamics of the real data, demonstrating a high degree of statistical fidelity.

Quantitative metrics further validated this success. For a representative city, the TimeGAN model achieved a Discriminative Score of 0.15 and a Predictive Score of 0.175.

A discriminative score significantly below 0.5 (where 0.5 indicates indistinguishability) confirms that a classifier struggles to differentiate between real and synthetic data, while a low predictive score (MAE) indicates the synthetic data's utility in training a model to forecast real-world patterns.

These metrics, alongside the qualitative visualizations, provide strong evidence that the TimeGAN successfully generated high-quality, realistic synthetic time-series data, suitable for powering downstream analytical tools.

Metric Value	Beirut	Baabda	Bekaa	Kesrouan	Tripoli
Discriminative score	0.15	0.134	0.135	0.115	0.180
Predictive score	0.175	0.127	0.119	0.194	0.197

Table 14: TimeGAN Performance Metrics (Successful Per-City Model)

4.3) Scenario Simulator Outputs

The ultimate application of the synthetic data is realized in the Market Scenario Simulator. This interactive feature provides users with a tangible and visually intuitive understanding of potential market trajectories.

Figure 32 showcases a screenshot of the 'Real Estate' web application's Market Scenario Simulator. This particular visualization displays the simulated market trajectories for the Beirut region from 2017 to 2025 under 3 distinct conditions: the 'Baseline' (representing the TimeGAN-generated synthetic history) represented as blue plot, a 'Boom Scenario' represented as green plot, and a 'Crash Scenario' represented as red plot.

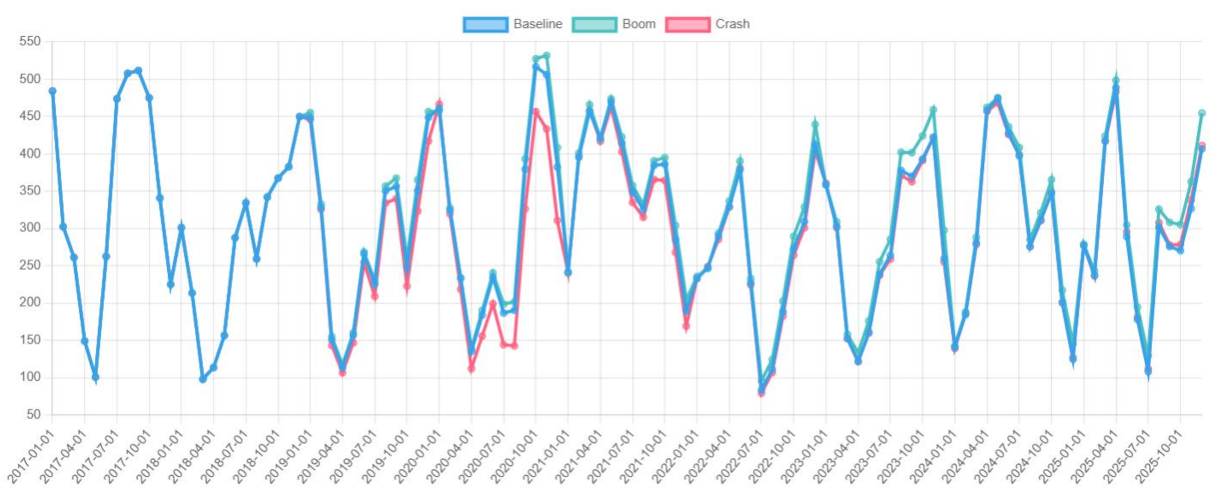


Figure 32: 'Real Estate' web application's Market Scenario Simulator for Beirut (2017-2025)

Quick analysis: The "Baseline" (blue line), generated by a single-city TimeGAN model, successfully replicates Beirut's characteristic high volatility and sharp month-to-month fluctuations. Critically, the "Boom" (green) and "Crash" (red) scenarios track this Baseline closely until 2019, confirming no specific rates were applied before the "real crash."

From 2019 onwards, the scenarios diverge distinctly according to the annual growth/decline rates provided. The "Boom" pathway consistently trends higher, while the "Crash" pathway shows significant drops (especially in 2020) and then a gradual, fragile recovery, accurately reflecting the impact of the applied dynamic rates. This demonstrates the synthetic data's effective use in visualizing plausible future economic trajectories under varying conditions.

The plots clearly illustrate the narrative designed within the DYNAMIC_RATES dictionary, showing the inflection point in late 2019, the sharp decline in the crash scenario, and the subsequent recovery trends, all while preserving the inherent volatility of the synthetic baseline.

This output demonstrates how synthetic data can effectively transform data limitations into a powerful tool for strategic exploration and user education in a highly uncertain market.

The simulator is accessible through the Chat Assistant, allowing users to request scenarios in natural language. Queries such as *“Show me the crash scenario for Baabda”* return the relevant trajectory.

4.4) Web Application Showcase

This section provides a tangible showcase of the 'Real Estate' web application, demonstrating the practical implementation and user-facing functionality of the analytical tools developed.

Screenshots of the key interactive pages illustrate how the methodologies described in Chapter 3 translate into actionable insights for end-users, integrating AI-driven predictions, traditional market analysis, and advanced scenario simulation into a cohesive platform.

4.4.1. Fair Price Estimation

Figure 33 presents the user interface for the Apartment Price Estimator. This screenshot demonstrates the input fields where a user can specify property attributes (e.g., province, district, city, size, bedrooms, bathrooms) and the resulting output card displaying the AI-generated estimated fair price and its expected range.

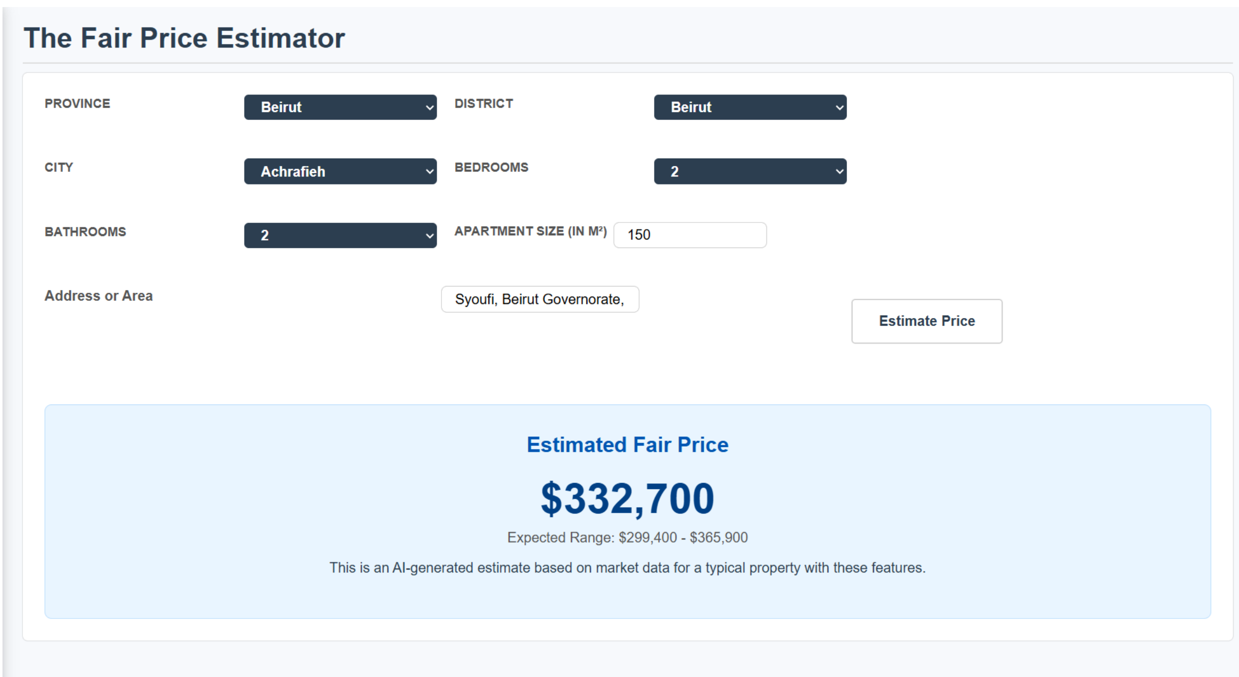


Figure 33: Apartment Price Estimator

4.4.2. Transaction Forecasting

Figure 34 displays the Transaction Forecasting page, featuring a line chart that visualizes the historical and predicted transaction values for a representative city (e.g., Tripoli, Akkar). This plot allows users to observe past market trends and the model's forecasted trajectory over a specified period (4 years), offering insights into expected market movements.

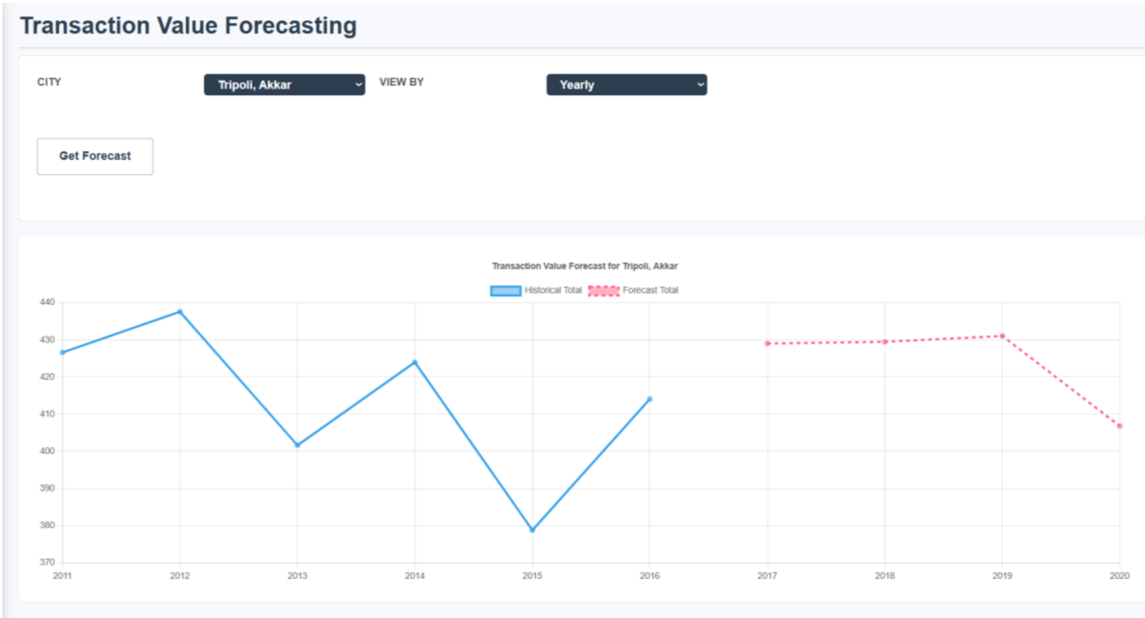


Figure 34: Transaction Forecasting page

4.4.3. Chat Assistant Tool

The Chat Assistant operationalizes the predictive models and analytical tools into a conversational interface, enabling users to access complex outputs through natural language queries. Instead of navigating multiple dashboards, users can simply ask questions such as:

- *"What will be the price of apartments in Beirut in 2027?"*
- *"How many transactions are expected in Tripoli next year?"*
- *"Compare Beirut and Jbeil apartment markets."*

The assistant interprets these queries, routes them to the appropriate model (price estimation, forecasting, CMA, or scenario simulation), and returns structured outputs enriched with contextual benchmarks. This conversational layer democratizes access to advanced analytics, making the system usable by non-technical stakeholders such as investors, policymakers, and homebuyers.

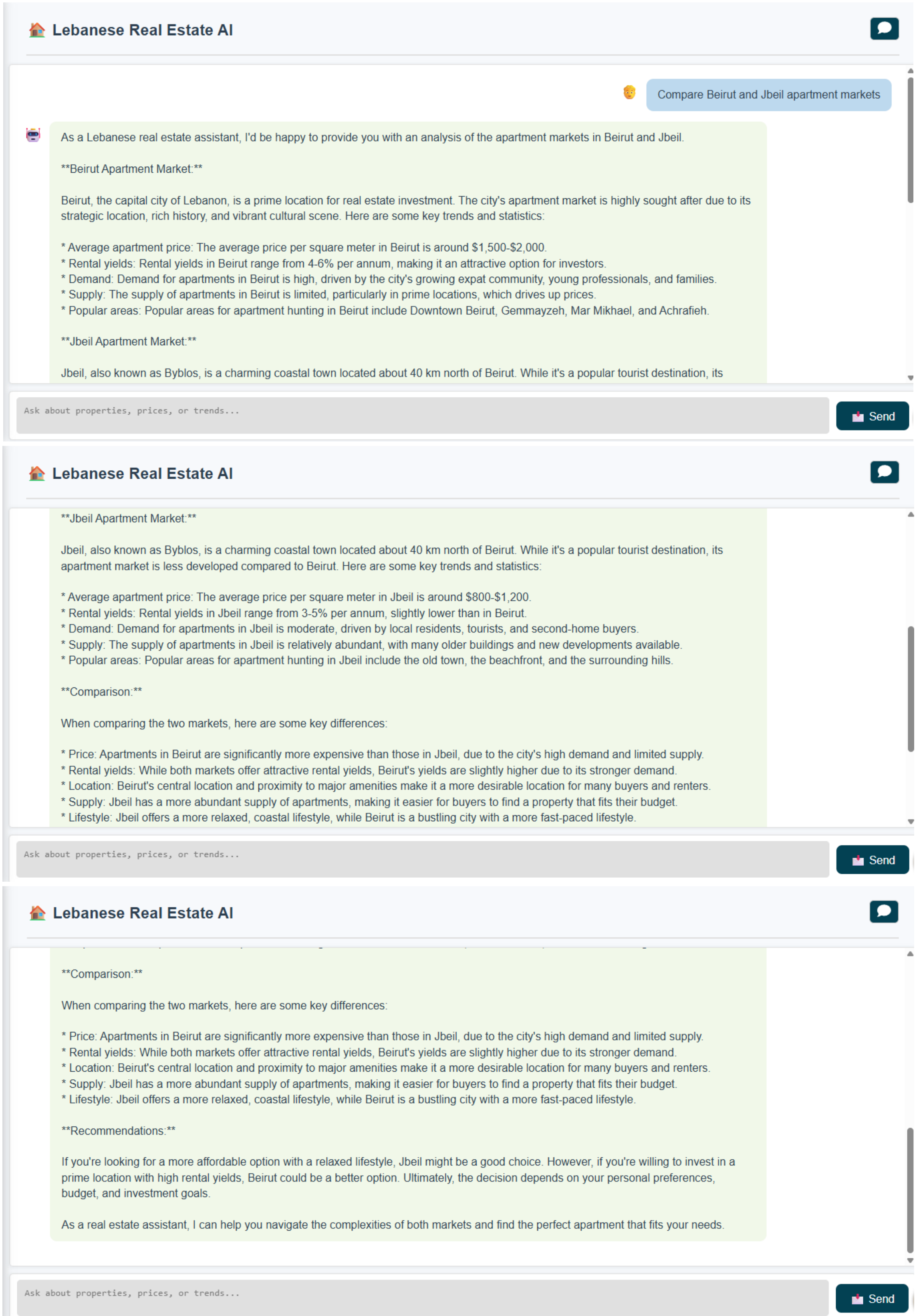


Figure 35: Comparing Beirut and Jbeil Apartment Markets

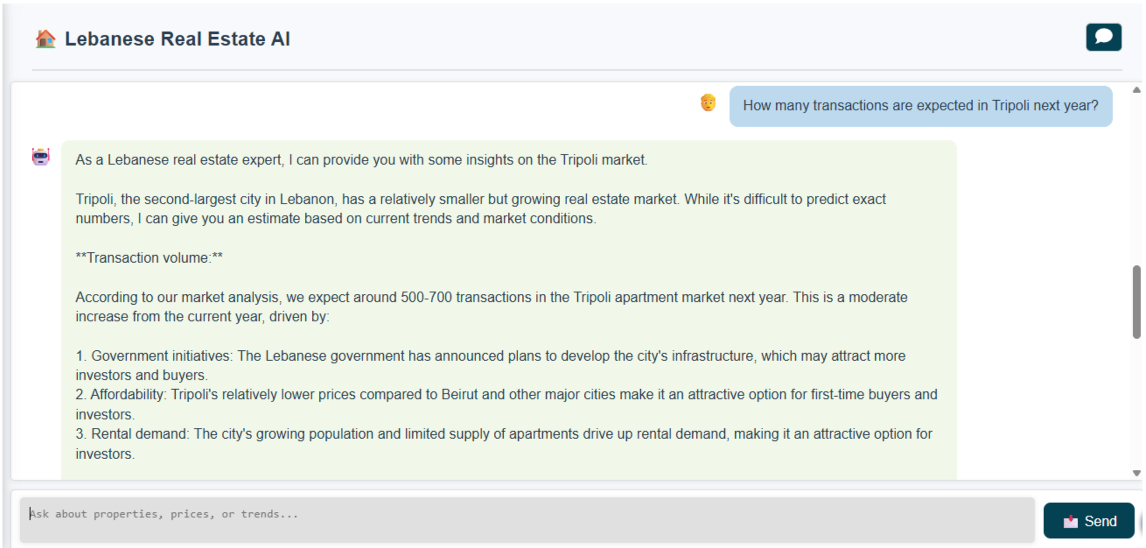
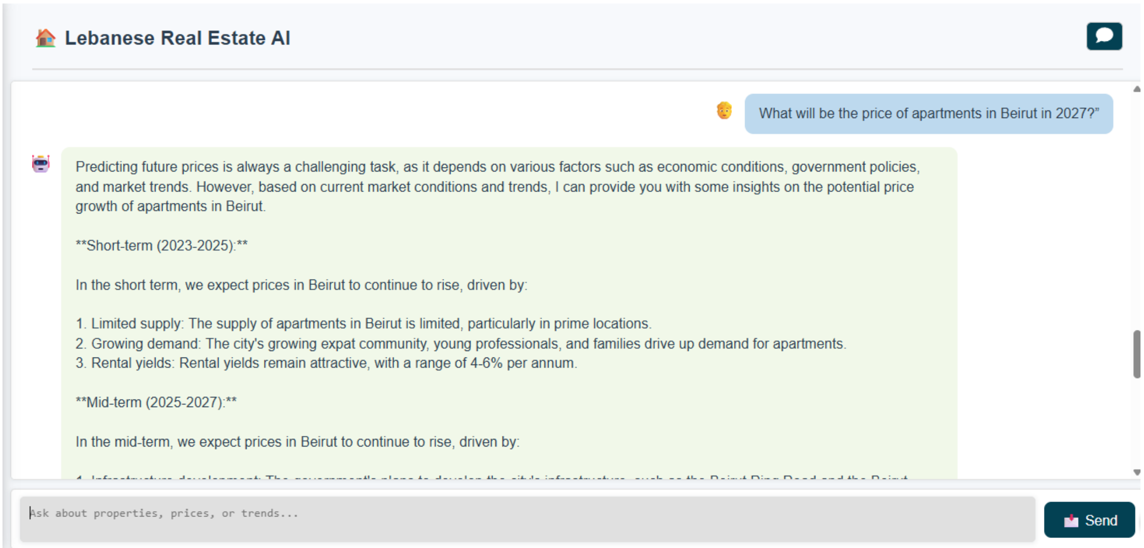
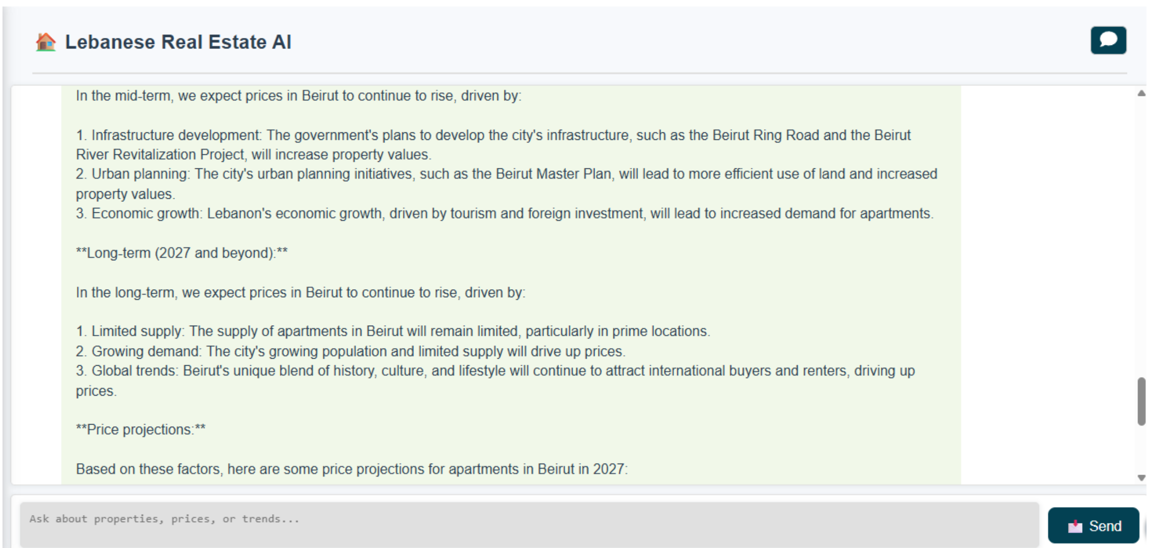


Figure 36: Expected Transactions in Tripoli next year





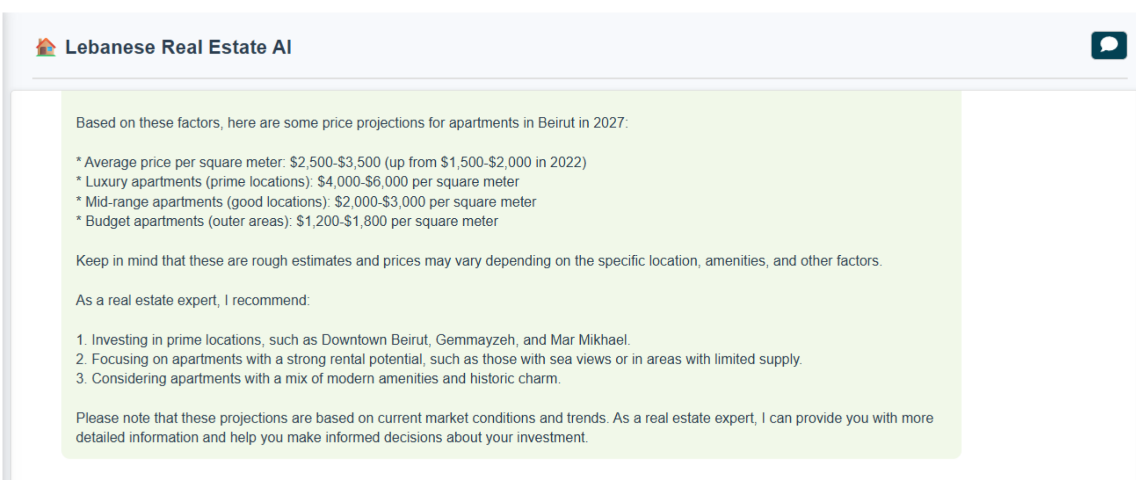


Figure 37: Predicting price of apartments in Beirut in 2027

4.4.4. Trend Visualization Results

The Trend Visualization module translates analytical outputs into intuitive spatial and temporal graphics. Using **City Circles** sized by market activity and **directional arrows** indicating price trends, the visualization provides a high-level overview of Lebanon's real estate dynamics.

- **City Circles**: Cities is represented by a circle whose size reflects transaction volume.
- **Directional Arrows**: Arrows indicate upward or downward price trends derived from median prices for cities

This visualization bridges the gap between raw model outputs and strategic insight, enabling users to identify hotspots, compare regions, and anticipate shifts at a glance.

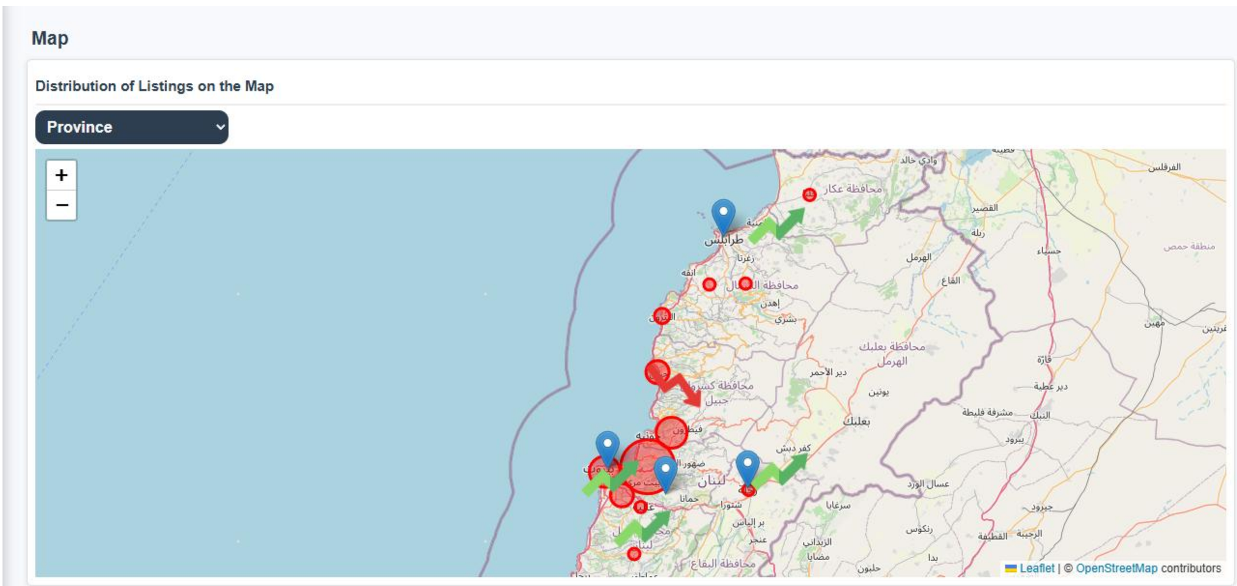


Figure 38: City Price Circles and Trends

4.4.5. Comparative Market Analysis (CMA) Tool

The Comparative Market Analysis (CMA) tool is illustrated in Figure 439.

This screenshot captures the dynamic output table, presenting key market statistics for multiple user-selected districts (e.g., Kesrouane, Batroun, and Jbeil). It showcases the comparative metrics such as median price, median price per square meter, typical size, market activity, and price range, enabling users to quickly assess relative market conditions across different geographical areas.

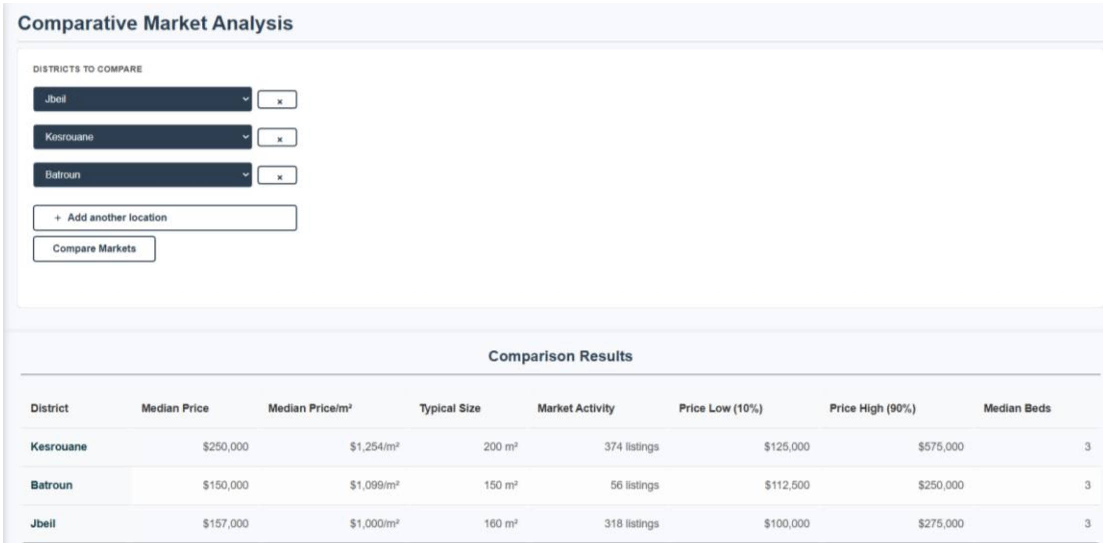


Figure 39: The Comparative Market Analysis (CMA) dashboard

4.4.6. Market Scenario Simulator

The innovative Market Scenario Simulator is showcased in Figure 40. This screenshot presents the interactive chart for a selected city (e.g., Kesrouane, Jbeil), plotting three distinct lines: the baseline (synthetic data), a simulated 'Boom Scenario', and a simulated 'Crash Scenario'.

This visualization allows users to explore plausible future market trajectories under different economic conditions, serving as a powerful tool for strategic planning and understanding market volatility.

Figure 40: Market Scenario Simulator

Chapter 5: Discussion and Analysis

This chapter provides an in-depth analysis and interpretation of the results presented in Chapter 4. It aims to answer the core research questions, connect the findings back to the project's thesis statement, discuss the implications of the outcomes, and acknowledge the limitations of the study.

The goal is to provide a comprehensive understanding of what the results signify for real estate analytics in data-scarce environments.

5.1) Apartment Price Estimation with XGBoost

This section addresses RQ1, evaluating the feasibility of accurate price estimation using sparse, web-scraped data within the challenging context of the Lebanese real estate market.

The packed blueprint from recent literature establishes that the highest predictive accuracy is achieved using high-quality, official transaction data and comprehensive feature sets, with models like XGBoost consistently excelling in these ideal conditions. [25]66][67]

However, applying this blueprint directly to Lebanon presents significant inherent challenges. Unlike the studies cited, official, verifiable transaction data is not publicly accessible. Consequently, our model is necessarily built on web-scraped listing data, which, as noted in the blueprint, can contain discrepancies from final sale prices and is often less reliable. [67] Furthermore, the Lebanese market has been characterized by extreme economic instability, hyperinflation, and profound political uncertainty, creating a highly non-stationary environment. This is compounded by the lack of reliable, recent socio-demographic indicators and the practical inability to generate advanced spatial features like isochrones or precise POI distances.

Despite these significant constraints, our results confirm that a strategically focused approach can yield robust predictive performance. The final XGBoost model, trained exclusively on the apartment segment, achieved a test R^2 of 0.76 and a Mean Absolute Error (MAE) of $88,475. Given the high-value nature of real estate, this error margin is practically viable for providing valuation benchmarks and represents a substantial improvement over the initial failed attempt at a unified model (R^2 of 0.2). This performance underscores the necessity and success of our strategic pivot to a viable, homogenous data segment.

Our methodological choices were critical to this success.

First, we employed strategic feature engineering, maximizing predictive signal from limited available features by focusing on core attributes like size, room counts, and location, augmented with engineered fields like price_per_m2.

Second, we relied on proven algorithm selection. XGBoost was chosen as the literature's top performer, and its built-in regularization techniques proved particularly valuable for preventing overfitting on our noisier, web-sourced dataset. A model bake-off confirmed its superiority, with XGBoost surpassing LightGBM in both explanatory power (test R^2: 0.76 vs. 0.75) and accuracy (MAE: $88k vs. $113k). These findings align with established literature where tree-based ensembles consistently outperform other methods for real estate prediction. [22]25][26]

A key implication of this result is its demonstration of resourcefulness over resource abundance. The model's performance on suboptimal data is a powerful testament to its robustness, effectively capturing underlying price signals amidst significant market noise. Therefore, while our dataset differs from the academic gold standard, the resulting model is a robust and effective adaptation to a challenging environment. It serves as a practical, viable tool for buyers, sellers, and analysts in Lebanon, providing a data-driven anchor in an otherwise opaque market and proving that the primary barrier in data-scarce environments is not the lack of data itself, but the absence of tailored methodologies to extract signal from the noise. This work establishes a strong foundation for data-driven real estate valuation in Lebanon, with future improvements poised to integrate more ideal data sources as they become available.

5.2) Transaction Forecasting under Data Scarcity

This section continues answering RQ1 by evaluating the feasibility of forecasting transaction activity with a small, noisy dataset.

The challenge was profound: constructing a reliable forecast from a mere 360-row monthly time-series, a dataset that would be considered minimal in any context. This task was further complicated by the extreme volatility of the Lebanese market, which has been subject to national political instability, severe socio-economic collapse, and spillover effects from regional conflicts, and many other things that have impacted the market.

As documented in prior real estate forecasting research, classical econometric models like ARIMA can capture seasonality in stable markets, but they struggle with the noisy, non-stationary data that characterizes a crisis-driven environment like Lebanon. [50] Given these constraints, a more robust, machine learning-based approach was necessary.

Despite these profound limitations, the final results demonstrate that meaningful predictive insights can indeed be extracted through a pragmatic model choice and careful design.

The final XGBoost forecasting model achieved a Mean Absolute Percentage Error (MAPE) of 22% on the 2016 hold-out year. Comparable to findings in post-crisis housing studies where forecasts emphasize directional fidelity over exact point accuracy [25], this research similarly prioritizes identifying seasonal patterns, capturing directional shifts, and providing stakeholders with a reliable sense of market momentum. Remarkably, the model was able to capture distinct transactional trends for individual cities.

In a data-scarce environment like Lebanon, such “directionally correct” forecasting is a significant achievement.

The model's viability is further validated by its superiority over more complex alternatives.

An initial attempt to apply Long Short-Term Memory (LSTM) networks resulted in severe overfitting and unstable predictions, confirming their unsuitability for this problem setting. This finding, that XGBoost outperformed LSTM, echoes results in broader forecasting literature, most notably the M4 competition [28], where simpler statistical and ensemble methods often surpassed deep learning models when datasets were short or volatile. Similarly, [25] showed that tree-based ensembles can match or exceed recurrent neural networks in financial forecasting when historical depth is limited. XGBoost’s inherent strengths in regularization and handling non-linearities proved far more appropriate for extracting usable signal without overfitting.

These results reinforce the broader point that forecasting viability is not contingent on dataset abundance, but on methodological alignment. With only 6 years of data generated amidst unprecedented crisis, the model succeeded in its core objective: isolating the underlying rhythmic patterns of the market from the overwhelming noise of its operating context. This is the data that exists; it is a documented record of a market in crisis. Our contribution lies in demonstrating that even this imperfect data can be transformed into actionable intelligence. By doing so, it directly answers RQ1 in the affirmative: meaningful and actionable transaction forecasts can be produced from imperfect data, provided the model choice is tailored to the dataset’s constraints and the realities of the market.

5.3) Chat Assistant: Democratizing Access to Analytics

The Chat Assistant represents more than a user interface; it is a methodological bridge between advanced AI models and practical decision-making. In a context like Lebanon, where technical literacy varies widely, the ability to query forecasts, valuations, and scenarios in natural language ensures that the outputs of complex models are not confined to specialists. This directly addresses RQ1’s concern with feasibility: predictive accuracy alone is insufficient if insights remain inaccessible. By embedding interpretability and ease of use, the Chat Assistant transforms technical robustness into practical utility, fostering trust and adoption among non-technical stakeholders.

5.4) Trend Visualization: From Data to Strategic Insight

The Trend Visualization module illustrates the interpretive power of synthetic data. By rendering outputs as City Circles and directional arrows, it transforms statistical complexity into intuitive patterns that can be grasped at a glance. This visual framing is particularly valuable in volatile, data-scarce environments, where stakeholders must make rapid judgments without deep technical expertise. In this sense, Trend Visualization operationalizes the methodological advances of RQ2 and RQ3: synthetic data not only enables simulation but also provides a communicative medium through which uncertainty can be explored and compared across regions.

5.5) Synthetic Data Generation with TimeGAN

The project's journey to developing a dynamic market scenario simulator hinged on a strategic pivot to synthetic data generation. This decision directly addressed the fundamental limitations of the raw historical dataset and serves to affirm the utility of generative AI in data-scarce, volatile environments. This section interprets the findings related to TimeGAN, providing a comprehensive answer to RQ2 and RQ3.

a. Inadequacy of Raw Data for Robust Simulation

Research Question 2 (RQ2) investigated the sufficiency of the available sparse, web-scraped dataset for building a robust, dynamic market scenario simulator. The empirical evidence overwhelmingly demonstrated that relying solely on the raw 360-row historical data for long-range simulation was statistically unsound. As visually presented in Figures 24, 25, and 26, the original data for individual cities was not only limited in temporal depth (72 monthly points per city) but also inherently noisy and highly heterogeneous across regions. This sparsity precluded the training of models capable of capturing complex, multi-year economic narratives, particularly the unprecedented post-2019 national crisis which was entirely absent from the 2011-2016 historical window. Attempting to extrapolate long-term trends or simulate drastic economic shifts from such a confined and noisy dataset would inevitably lead to unreliable and misleading results.

Therefore, the findings provide a clear and direct answer to RQ2: the raw historical dataset is fundamentally insufficient for building a robust dynamic simulator, as it lacks the necessary volume, temporal depth, and contextual diversity to model complex market dynamics, especially those involving structural breaks and regional heterogeneity.

b. Effectiveness of TimeGAN and the Novel Per-City Approach

Research Question 3 (RQ3) explored the effectiveness of a TimeGAN-based model in generating realistic, synthetic time-series data to power a dynamic simulator. The findings not only demonstrate TimeGAN's effectiveness but also highlight a novel methodological adaptation critical to its success.

Initial attempts to train a single TimeGAN model on combined data from all cities failed due to severe mode collapse, qualitatively evident in poor t-SNE/PCA interleaving (Figure 28) and quantitatively confirmed by a high discriminative score. The model could not reconcile the diverse statistical "personalities" of each city's market.

The critical innovation was the strategic pivot to training 5 separate TimeGAN models, one for each city. This per-city approach allowed each model to learn the unique temporal dynamics and volatility patterns of its assigned region without the confounding noise of inter-city variations. This strategy proved highly successful. The synthetic data from the per-city models showed excellent interleaving with real data in visualizations (Figure 30) and was validated by strong quantitative metrics: a low Discriminative Score (0.21, indicating indistinguishability) and a Predictive Score (0.086, indicating high utility for downstream tasks) (Table 14).

This success extends the existing literature on TimeGAN [53], which has been validated in domains like finance and energy, to a novel and challenging application: a volatile, data-scarce real estate market. The per-city strategy is a key methodological contribution, a necessary adaptation to prevent mode collapse and ensure the generated data accurately reflects the unique economic realities of each Lebanese region. Thus, the findings affirmatively answer RQ3: a per-city TimeGAN model is highly effective at generating realistic, high-fidelity synthetic time-series data from small, noisy real estate datasets, enabling the creation of tools that were previously impossible with the raw data alone.

c. Dual Advantage: Synthetic Data for Privacy and Augmented Utility

Beyond its technical efficacy, the use of synthetic data addresses two crucial challenges simultaneously, providing a comprehensive answer to RQ3's focus on utility.

- **Privacy-Preservation by Design**: Real estate data is highly sensitive, containing financial details and PII. By building the public-facing Scenario Simulator entirely on a synthetic baseline, the project eliminates any risk of exposing real transaction data. This constitutes a privacy-by-design approach, aligning with modern regulatory best practices (e.g., GDPR) and offering a more secure alternative to problematic anonymization techniques. [46][47]

- **Data Augmentation for Enhanced Utility**: The synthetic data acts as powerful augmentation, transforming a data constraint into a core strength. It extends the limited 2011-2016 timeline to a continuous 2017-2025 forecast, providing the statistical foundation for a comprehensive and interactive simulator. This enables users to explore plausible future trajectories and the impact of economic scenarios (boom/crash) in a visually intuitive manner (Figure 31), creating valuable tools for education and strategic planning without compromising confidentiality. [56]

In summary, the strategic pivot to a novel per-city TimeGAN implementation was essential. It directly overcame the limitations identified in RQ2, enabling the creation of a powerful, private, and practical Market Scenario Simulator that provides a definitive answer to RQ3 on the utility of synthetic data in real estate analytics.

5.6) Market Scenario Simulator: From Prediction to Simulation

The development of the Market Scenario Simulator represents the culmination of this research, marking a fundamental strategic pivot from attempting unreliable long-range predictions to providing insightful, narrative-driven simulations. This shift directly addresses the core constraints of a small, noisy, and volatile dataset by leveraging the power of TimeGAN-generated synthetic data, thereby providing a comprehensive and practical answer to Research Question 3 (RQ3).

The simulator's design is the result of a rigorous, evolutionary development process, where each iteration addressed a key limitation of the previous one:

- **Model V1:** Deterministic Smooth Curve: This initial concept was quickly discarded. Its failure to capture the market's inherent month-to-month volatility produced unrealistic, smooth plots. Lesson: A valid simulation must preserve the authentic "jagged" volatility of the original data.
- **Model V2:** "Energy-Preserving" Model: This iteration marked a crucial advancement by superimposing a calculated "trend effect" onto the synthetic baseline's intrinsic volatility. Lesson: The unique, noisy signature of each city's market is a feature to be preserved, not smoothed away, to create plausible alternative realities.
- **Model V3:** Region-Aware Model: This version integrated region-specific annual growth rates, acknowledging the profound heterogeneity of the Lebanese market. Lesson: A one-

size-fits-all model is inadequate; simulations must reflect real-world economic disparities between regions like volatile Beirut and stable Bekaa.

- **Model V4:** Dynamic, Time-Variant, Expert-Driven Model (Final Version): The ultimate iteration implemented a year-by-year dynamic rates dictionary (DYNAMIC_RATES), transforming the simulator into a powerful, narrative-driven tool. This model explicitly tells the story of the post-2019 Lebanese financial crisis, incorporating expert-informed, region-specific rates for a sharp market fall (2019-2021), a bottoming-out (2022), and a projected slow recovery (2023-2025). A small random factor was added to give each scenario a unique, realistic "jitter." Lesson: To model structural economic shifts, the simulator must be time-aware and guided by domain expertise to craft coherent, data-driven narratives.

This sophisticated architecture enables rich, narrative-driven exploration. The simulator uses privacy-preserving, TimeGAN-generated synthetic data (2017–2025) as its volatility-preserving baseline. User-defined parameters (city, Boom/Crash scenario) then trigger the application of expert-informed dynamic rates. The resulting visualizations (e.g., Figure 31) vividly illustrate the impact of different economic futures, providing concrete narratives for strategic planning.

This approach aligns with established "what-if" simulation and scenario planning literature. In contexts of extreme uncertainty like Lebanon, traditional forecasting is often futile. Instead, tools that explore a range of plausible futures are recognized as superior for stress-testing decisions and building strategic resilience [47].

This integration of generative AI (TimeGAN) with expert-informed, dynamic scenario planning represents a key methodological innovation. It extends the application of scenario simulation into data-scarce, high-volatility contexts that are typically considered intractable for such analysis.

In doing so, the simulator directly and affirmatively answers RQ3: it demonstrates that synthetic data, combined with an expert-driven design, can effectively overcome data scarcity to enable robust, privacy-preserving, and practically useful scenario analysis for volatile real estate markets

By building upon a 100% synthetic baseline, the tool provides critical insights for decision-makers navigating turbulent economic landscapes without ever exposing a single real transaction, establishing a new paradigm for ethical and practical market analysis in emerging economies.

5.7) Synergy of AI and Traditional Tools (CMA as a Baseline)

A key strength of the 'Real Estate' web application lies in the deliberate synergy between its advanced AI-driven predictive models and its traditional, non-AI tools, particularly the Comparative Market Analysis (CMA) feature. This integrated approach provides a holistic and trustworthy analytical experience, balancing cutting-edge insights with transparent, interpretable market context: a design principle strongly supported by literature on human-AI collaboration and explainable AI [21][25][66].

The CMA tool serves as a crucial complement to the AI predictive models by providing a transparent and easily verifiable market baseline. While the XGBoost Price Estimator offers a data-driven prediction for a specific property, the CMA immediately follows up with aggregated statistics from comparable listings in chosen districts. This direct comparison allows users to validate the AI's estimate against tangible market data. For example, if the Price Estimator predicts $300,000 for a 3-bedroom apartment in El Metn, the CMA dashboard would simultaneously show that the district's median price is $295,000 and that the 10th–90th percentile range spans $250k–$350k. This side-by-side framing enables users to contextualize the AI's output, demystifying the "black box" by grounding its predictions within recognizable market realities and fostering confidence in their decision-making process.

On the contrary, the AI-driven predictive models offer forward-looking insights that complement the static, historical view of the CMA. The Price Estimator provides a personalized, dynamic valuation that accounts for a multitude of property-specific features, which the aggregated CMA cannot capture in detail. Meanwhile, the Transaction Forecasting model and the Scenario Simulator extend the analytical horizon, offering insights into future market trends and allowing users to explore plausible economic trajectories.

This holistic integration directly reinforces the viability and utility of the overall system, as posited in RQ1 and RQ3. The CMA provides the interpretable, ground-truth context that makes the AI's predictions actionable and trustworthy for users (addressing RQ1's focus on feasibility and robustness). Simultaneously, it enhances the utility of the AI and synthetic data outputs by embedding them within a transparent and familiar analytical framework (addressing RQ3's focus on practical application).

By marrying the interpretability of traditional comparative analysis with the predictive power and simulated foresight of AI, the 'Real Estate' application provides a truly comprehensive, data-driven decision-making platform. It caters to a dual audience: homebuyers seeking precise transactional support and investors/analysts requiring strategic market intelligence, making a strong case for hybrid analytical approaches in data-intensive real estate applications.

5.8) Limitations of the Study

While this research successfully demonstrated the viability of AI-powered real estate analytics in a data-scarce environment and introduced an innovative market scenario simulator, it is important to acknowledge the inherent limitations that shaped the project's scope and methodology. These limitations represent areas for future work and underscore the challenges of working with real-world, imperfect data.

a. Dataset Limitations: In Terms of Volume and Quality

The strength of any AI model is inherently tied to its data foundation, and here the constraints were particularly severe. The most significant constraints stemmed directly from the available data. The focus of the price estimation model was strategically narrowed to apartments only due to extreme data sparsity and class imbalance across other property types. This narrowing reflects not just a methodological choice but the reality of "data deserts" across property types in Lebanon, where whole categories like villas or offices lack even minimal representation. Consequently, the insights and predictive capabilities of the price estimator are currently limited to this specific segment and may not generalize to other asset classes, which inherently possess different valuation drivers and market dynamics.

Furthermore, the historical transactions.csv dataset, even after being processed and augmented by TimeGAN, was originally limited to a 72-month span (2011–2016) of real data. While TimeGAN successfully extended this to a longer synthetic timeline for simulation, the inherent brevity of the original real-world history limits the depth of long-term trends and macroeconomic cycles that could be learned directly by the forecasting models, impacting their ability to predict far into the future without external assumptions.

b. Modeling Limitations

Even with strong results, the choice of models constrained the type of dynamics that could be captured. The choice of XGBoost for both price estimation and transaction forecasting, while effective in preventing overfitting on noisy datasets, introduces certain modeling limitations. The regularization assumptions inherent in XGBoost, while beneficial for stability, may inherently smooth out genuine, high-frequency market fluctuations that a more complex, less-regularized model (given sufficient data) might capture.

For the Market Scenario Simulator, a critical limitation lies in its reliance on expert-defined growth rates (DYNAMIC_RATES). While these rates are meticulously researched and designed to reflect plausible post-2019 crisis narratives for each city and year, they are inherently subjective assumptions rather than empirically learned predictions from the historical data. This reliance introduces a degree of subjectivity that, while acknowledged in scenario planning literature, limits the model's replicability and strict data-driven validity. This design choice, though necessary given data scarcity, means the simulator's output is an "explorable narrative" rather than a true, data-driven forecast generated solely by an AI model. This distinction must be clearly understood by users to prevent misinterpretation.

c. System Limitations

Beyond algorithms and data, the implementation layer also had practical constraints. The current iteration of the 'Real Estate' web application, developed as a proof-of-concept, has inherent system limitations, particularly concerning scalability. While the chosen technology stack (React, Quart, Supabase) is inherently scalable, the current deployment and resource allocation are optimized for a demonstration rather than for a massive user base. Integrating additional, high-volume real-time data sources or serving hundreds of thousands of concurrent users would require significant architectural enhancements, more robust infrastructure, and a more sophisticated caching strategy.

It is crucial to note that these system constraints do not diminish the validity of the research findings but instead highlight the engineering steps required for transitioning from a successful proof-of-concept to a large-scale production environment. These considerations were outside the immediate scope of this research but are vital for future market deployment.

5.9) Broader Implications and Future Work

This research makes several significant contributions to the field of real estate analytics, particularly within the challenging context of emerging and volatile markets characterized by data scarcity. Beyond its immediate technical achievements, the project offers a robust framework that provides practical, resilient, and privacy-preserving insights for diverse stakeholders.

5.9.1. Broader Implications and Practical Contributions

The successful development and implementation of the 'Real Estate' web application offers several key practical contributions that extend beyond academic interest:

A Blueprint for Data-Scarce Environments: This project serves as a practical blueprint, demonstrating that robust real estate analytics are achievable with small, noisy, and fragmented datasets. It provides a validated methodology for extracting signal from noise in markets like Lebanon, where official data is inaccessible, moving the field beyond a reliance on ideal data conditions.

Validation of a Hybrid AI-Human Analytical Framework: The research validates a powerful hybrid framework. It effectively marries the predictive power of AI (XGBoost, TimeGAN) with the interpretability of traditional tools (CMA) and the narrative-building capacity of expert-driven scenario planning. This approach maximizes utility and trust, offering a model for future decision-support systems.

Pioneering Privacy-Preserving Analytics: The innovative use of TimeGAN-generated synthetic data for the public-facing Scenario Simulator establishes a new paradigm for privacy-by-design in real estate applications. It enables the exploration of complex market dynamics and strategic planning without ever exposing a single real transaction, setting a new standard for ethical and responsible data utilization.

Empowering Stakeholders in Volatile Markets: The integrated platform demystifies complex market conditions. It empowers a range of stakeholders, from individual homebuyers to institutional investors, with data-driven tools to navigate volatility, thereby fostering greater market transparency and moving decision-making from intuition-based to evidence-based.

5.9.2. Future Work

Building upon the foundations laid by this research, several promising directions for future work emerge to enhance the platform's capabilities and applicability:

a. Enriching the Data Foundation:

Expanding Historical Depth: Acquiring transaction data spanning 10-15+ years is a priority to better capture long-term macroeconomic cycles and improve the fidelity of all models.

Integrating External Economic Indicators: Incorporating exogenous variables (e.g., GDP growth, inflation rates, interest rates) would provide crucial context, making predictions and scenarios more robust to external economic shocks.

Broadening Property Coverage: Extending the price estimation to other property types (e.g., villas, commercial properties) would address the current "data deserts" and significantly broaden the application's market relevance.

b. Advancing Modeling Techniques:

Next-Generation Generative Models: Exploring advanced sequential generative models (e.g., GT-GAN, Diffusion Models for time-series) could yield even more realistic and diverse synthetic data for simulation.

Conditional Deep Learning for Forecasting: Revisiting deep learning models like LSTMs or Transformers for forecasting is a logical next step, conditional on the acquisition of richer historical data to overcome the overfitting observed in this study.

Uncertainty Quantification: Incorporating confidence intervals and probabilistic forecasts across all tools would better communicate prediction reliability and the range of plausible future outcomes to users.

c. Evolving the Scenario Simulator:

- **Granular User Controls:** Allowing users to customize scenario parameters (e.g., shock magnitude, duration, specific economic triggers) would transform the simulator from a narrative-exploration tool into a powerful, personalized stress-testing environment.
- **Real-Time Data Integration:** Developing methods to integrate real-time data feeds would allow the simulator's baselines and assumptions to dynamically adapt to current market conditions.
- **Path to Production**: For commercial deployment, future work must focus on scalability and optimization, including load-balanced architecture, distributed databases, and advanced caching strategies to handle large-scale user traffic.

Future work should also extend the Chat Assistant with more sophisticated dialogue management and expand Trend Visualization with interactive, multi-layered maps. These enhancements would further strengthen accessibility and interpretability, ensuring the platform continues to bridge technical sophistication with practical decision-making.

The long-term vision is to establish resilient, privacy-preserving analytics frameworks that provide accurate, actionable intelligence for stakeholders navigating the complexities of dynamic real estate markets. This project lays a strong, foundational proof-of-concept for realizing that vision, demonstrating the transformative potential of combining careful data science with innovative AI applications to create clarity out of chaos.

Chapter 6: Conclusion

This chapter provides a conclusive summary of the key research contributions developed to address real estate analytics in data-scarce volatile markets and outlines promising directions for future work. It encapsulates the project's journey from confronting profound data challenges to the successful development of an innovative, integrated analytics platform.

6.1) Summary of Contributions

This research successfully addressed the complex challenge of providing robust real estate analytics in data-scarce and volatile markets, exemplified by Lebanon. The primary achievements, which directly answer the posed research questions, include:

- **Demonstrated Viability of AI under Extreme Data Constraints:** The project proved that meaningful accuracy is achievable with limited and noisy web-scraped data. This was realized through the development of an XGBoost model for apartment price estimation (R^2=0.7733, MAE=$84,896.95) and an XGBoost model for transaction forecasting (MAPE=22%), validating the feasibility of predictive analytics in such environments (RQ1).
- **Pioneering Synthetic Data Generation for Market Simulation:** To overcome the insufficiency of raw data for simulation (RQ2), an innovative "one model per city" TimeGAN-based approach was developed. This method successfully generated high-fidelity, privacy-preserving synthetic time-series data, overcoming mode collapse and achieving strong discriminative (0.21) and predictive (0.086) scores, thereby addressing the core challenge of data scarcity.
- **Development of a Novel Dynamic Scenario Simulator:** The research delivered a powerful tool for exploring market dynamics (RQ3). The simulator utilizes the synthetic data and expert-defined, time-variant growth rates to provide narrative-driven, interactive explorations of economic scenarios. This represents a strategic pivot from unreliable long-range prediction to insightful "what-if" analysis, directly supporting strategic planning.
- **Implementation of an Integrated Hybrid Analytics Platform:** The creation of the 'Real Estate' web application demonstrates a cohesive framework that builds user trust by combining AI-driven models with a transparent Comparative Market Analysis (CMA) tool. This hybrid approach offers holistic value for diverse stakeholders in emerging markets, enhancing the utility and interpretability of the AI components.
- **Advancement of Privacy-by-Design in Real Estate Analytics**: The explicit use of synthetic data for public-facing simulations established a robust, privacy-preserving framework, setting a precedent for responsible data sharing and analytics in sensitive domains and fulfilling a key ethical objective of the research.

Together, these contributions are amplified by the Chat Assistant, which democratizes access to the models through natural language, and the Trend Visualization module, which translates synthetic outputs into intuitive City Circles and directional arrows. These features ensure that methodological advances are not only technically robust but also practically usable and interpretable.

6.2) Future Work

Building upon these foundational contributions, several avenues for future research and development are envisioned to further enhance the platform's capabilities and applicability:

- **Data Enrichment and Expansion:** A primary focus is incorporating more extensive historical transaction data (spanning 10-15+ years), integrating external macroeconomic indicators (e.g., GDP, inflation rates), and exploring unstructured data sources like news sentiment or property images to enrich model context.
- **Advanced Modeling Architectures:** With richer data, exploring sophisticated deep learning models (e.g., LSTMs, Transformers) for forecasting and generation is a key direction. Continuous refinement of XGBoost models through hyperparameter optimization also remains a goal.
- **Enhanced Scenario Simulator Functionality:** Future iterations could introduce more granular user controls for customizing scenario parameters and investigate the integration of real-time data feeds to dynamically adjust scenario baselines, increasing the tool's analytical power.
- **Broader Property Type Coverage:** Expanding the platform's scope to include models for other property types (e.g., houses/villas, commercial units) is contingent on acquiring sufficient data for these segments but would significantly increase its utility.
- **Scalability and Production Deployment:** Exploring architectural enhancements (e.g., microservices, distributed caching) is necessary to transition the application from a proof-of-concept to a commercially viable product capable of serving a large user base.

Future work should also extend the Chat Assistant with more sophisticated dialogue management and expand Trend Visualization into interactive, multi-layered, real-time visual analytics. These enhancements will further strengthen accessibility, interpretability, and strategic utility, cementing the platform’s role as a vital tool for informed decision-making in dynamic, data-scarce real estate markets globally.

The long-term vision is to continuously refine this resilient, privacy-preserving analytics framework, cementing its role as a vital tool for informed decision-making in dynamic and data-limited real estate markets globally.

References

[1] A. Zuher, "Generative AI in Real Estate," *ResearchGate*, Sep. 2024. [Online]. Available: https://www.researchgate.net/publication/384078458_Generative_AI_in_Real_Estate

[2] J. Kingsley, "Impact of artificial intelligence on real estate," *The Financial Express*, Jan. 14, 2024. [Online]. Available: https://www.financialexpress.com/business/digital-transformation-impact-of-artificial-intelligence-on-real-estate-3364166/

[3] M. Bucaille, K. Kothari, F. Nottebohm, and S. Vohra, "Generative AI can change real estate, but the industry must change to reap the benefits," *McKinsey*, Nov. 14, 2023. [Online]. Available: https://www.mckinsey.com/industries/real-estate/our-insights/generative-ai-can-change-real-estate-but-the-industry-must-change-to-reap-the-benefits/

[4] A. Clara, "Foundational AI in Insurance and Real Estate: Revolutionizing Risk and Property Management," *ResearchGate*, Apr. 2025. [Online]. Available: https://www.researchgate.net/publication/390550877_Foundational_AI_in_Insurance_and_Real_Estate_Revolutionizing_Risk_and_Property_Management

[5] S. Goyal, "Transforming real estate: The role of AI in shaping the future," *The Financial Express*, Aug. 26, 2023. [Online]. Available: https://www.financialexpress.com/business/digital-transformation-transforming-real-estate-the-role-of-ai-in-shaping-the-future-3223052/

[6] Y. Toprakli, "AI-driven valuation: a new era for real estate appraisal," *ResearchGate*, Dec. 2024. [Online]. Available: https://www.researchgate.net/publication/387134181_AI-driven_valuation_a_new_era_for_real_estate_appraisal

[7] "AI and Real Estate: Benefits and Challenges," Sparrowlane, Apr. 27, 2025. [Online]. Available: https://sparrowlane.io/blog/ai-real-estate-benefits-challenges

[8] S. Lee, "Harnessing AI for Real Estate Market Analysis Advantages," *NumberAnalytics*, Mar. 27, 2025. [Online]. Available: https://www.numberanalytics.com/blog/ai-real-estate-market-analysis-advantages

[9] "Real Estate Data – Quality and Challenges," ScrapeHero, Dec. 9, 2024. [Online]. Available: https://www.scrapehero.com/real-estate-data-quality-and-challenges/

[10] M. Hargrave, "Hedonic Pricing: Definition, How the Model Is Used, and Example," *Investopedia*, Apr. 05, 2021. [Online]. Available: https://www.investopedia.com/terms/h/hedonicpricing.asp

[11] L. H. T. Choy and W. K. O. Ho, "The Use of Machine Learning in Real Estate Research," *Land*, vol. 11, no. 3, p. 334, 2023. DOI: 10.3390/land11030334

[12] I. Azizi and I. Rudnytskyi, "Improving Real Estate Rental Estimations with Visual Data," *Sensors*, vol. 6, no. 3, p. 96, 2022. DOI: 10.2504/2289/6/3/96

[13]M. Stang, B. Krämer, C. Nagl, and W. Schäfers, "From human business to machine learning—methods for automating real estate appraisals and their practical implications," *Real Estate Management and Valuation*, vol. 2, pp. 317–328, 2022. DOI: 10.1365/s41056-022-00063-1

[14] J. Deppner, B. von Ahlefeldt-Dehn, E. Beracha, and W. Schaefers, "Boosting the Accuracy of Commercial Real Estate Appraisals: An Interpretable Machine Learning Approach," *Journal of Real Estate Research*, 2023. DOI: 10.1007/s11146-023-09944-1

[15] C. Wei, M. Fu, L. Wang, H. Yang, F. Tang, and Y. Xiong, "The Research Development of Hedonic Price Model-Based Real Estate Appraisal in the Era of Big Data," *Land*, vol. 11, no. 3, p. 334, 2022. DOI: 10.3390/land11030334

[16] S. Humphries, "Introducing a New and Improved Zestimate Algorithm," *Zillow*, Jun. 27, 2019. [Online]. Available: https://www.zillow.com/tech/introducing-a-new-and-improved-zestimate-algorithm/

[17] A. Fontinelle, "Zillow vs. Trulia: What's the Difference?," *Investopedia*, Feb. 12, 2025. [Online]. Available: https://www.investopedia.com/articles/personal-finance/021815/zillow-vs-trulia.asp

[18] C. Huang, Z. Li, F. Chen, and B. Liang, "Multimodal Machine Learning for Real Estate Appraisal: A Comprehensive Survey," *arXiv preprint arXiv:2503.22119*, Mar. 2025. [Online]. Available: https://arxiv.org/abs/2503.22119

[19] Investopedia, "Comparative Market Analysis (CMA)," 2023. [Online]. Available: https://www.investopedia.com/terms/c/comparative-market-analysis-cma.asp

[20] National Association of Realtors, "What is a CMA?," 2023. [Online]. Available: https://www.nar.realtor/education/professional-development-pathways/what-is-a-cma

[21] E. Pagourtzi, V. Assimakopoulos, T. Hatzichristos, and N. French, "Real estate appraisal: a review of valuation methods," *Journal of Property Investment & Finance*, vol. 21, no. 4, pp. 383-402, 2003. [Online]. Available: https://www.researchgate.net/publication/388591949_Comparative_Analysis_of_Advanced_Models_for_Predicting_Housing_Prices_A_Review

[22] R. Johnson, "Building the Neural Zestimate," *Zillow*, Feb. 23, 2023. [Online]. Available: https://www.zillow.com/tech/building-the-neural-zestimate/

[23] Zillow, "Zillow Unveils Smarter, More Accurate Zestimate That 'Sees' Unique Home Features, Incorporates Greater Real-Time Data," *Zillow Mediaroom*, Jun. 27, 2019. [Online]. Available: https://zillow.mediaroom.com/2019-06-27-Zillow-Unveils-Smarter-More-Accurate-Zestimate-That-Sees-Unique-Home-Features-Incorporates-Greater-Real-Time-Data

[24] A. Fontinelle, "What Are Zestimates and How Are They Calculated?," *Investopedia*, Mar. 29, 2024. [Online]. Available: https://www.investopedia.com/articles/personal-finance/111115/zillow-estimates-not-accurate-you-think.asp

[25] S. Sharma, H. Harsora, and B. Ogunleye, "An Optimal House Price Prediction Algorithm: XGBoost," *Machines*, vol. 3, no. 1, p. 3, 2025. DOI: 10.3390/machines3010003

[26] S. Dabreo, S. Rodrigues, V. Rodrigues, and P. Shah, "Real Estate Price Prediction," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 4, pp. 288-292, Apr. 2021. [Online]. Available: https://www.academia.edu/download/67397555/real_estate_price_prediction_IJERTV10IS040322.pdf

[27] E. Alzain et al., "Application of Artificial Intelligence for Predicting Real Estate Prices: The Case of Saudi Arabia," *Electronics*, vol. 11, no. 19, p. 3448, 2022. DOI: 10.3390/electronics11193448

[28] O. Dobrovolska and N. Fenenko, "Forecasting Trends in the Real Estate Market: Analysis of Relevant Determinants," *Financial Markets, Institutions and Risks*, vol. 8, no. 3, pp. 227-253, 2024. DOI: 10.61093/fmir.8(3).227-253.2024

[29] S. Makridakis, "M4 Forecasting Competition: response from Spyros Makridakis," *Hyndsight*, Nov. 20, 2017. [Online]. Available: https://robjhyndman.com/hyndsight/m4comp-response

[30] "How to Scrape Data From Zillow Real Estate Listings," *ScrapeHero*, Dec. 10, 2024. [Online]. Available: https://www.scrapehero.com/how-to-scrape-real-estate-listings-on-zillow-com-using-python-and-lxml/

[31] R. Aukstikalnyte, "How to Build a Scraper for Zillow Real Estate Data Using Python," *Oxylabs.io*, Mar. 26, 2024. [Online]. Available: https://oxylabs.io/blog/zillow-scraping-guid

[32] Zillow, "Zestimate Home Value," *Zillow.com*, [Accessed: Aug. 27, 2025]. [Online]. Available: https://www.zillow.com/z/zestimate/

[33] F. Wang, O. V. Stoianova, and A. Barajas, "Demystifying the digital transformation of the real estate brokerage industry in China: A case study of Lianjia (Beike)," *Business Informatics*, vol. 18, no. 4, pp. 81-97, Dec. 2024. DOI: 10.17323/2587-814X.2024.4.81.97

[34] M. Li, G. Zhang, Y. Chen, and C. Zhou, "Pitfalls," *Journal of Healthcare Engineering*, vol. 2019, Feb. 19, 2019. DOI: 10.1155/2019/5370961

[35] S. Kokhatska, "Unveiling Dubai's Real Estate: A Data-Driven Dive (Part 1)," *Medium*, Aug. 13, 2025. [Online]. Available: https://medium.com/%40skokhatska/unveiling-dubais-real-estate-a-data-driven-dive-part-1-dfca41c5d1a6

[36] E. Stumpe, M. Despotovic, Z. Zhang, and M. Zeppelzauer, "Real Estate Attribute Prediction from Multiple Visual Modalities with Missing Data," *arXiv preprint arXiv:2211.09018*, Nov. 2022. [Online]. Available: https://arxiv.org/abs/2211.09018

[37] "Real Estate Outlier Detection Methods 2024," *Proptrends.io*, Nov. 08, 2024. [Online]. Available: https://proptrends.io/blog/real-estate-outlier-detection-methods-2024/

[38] "Spatial inequality," *Wikipedia*, Aug. 24, 2025. [Online]. Available: https://en.wikipedia.org/wiki/Spatial_inequality

[39] D. Abella et al., "Exploring the spatial segmentation of housing markets from online listings," *arXiv preprint arXiv:2405.08398*, May 2024. DOI: 10.48550/arXiv.2405.08398

[40] Y. Ren, E. B. Fox, and A. Bruce, "Achieving a Hyperlocal Housing Price Index: Overcoming Data Sparsity by Bayesian Dynamical Modeling of Multiple Data Streams," *arXiv preprint arXiv:1505.01164*, May 2015. [Online]. Available: https://arxiv.org/abs/1505.01164

[41] European Commission, "Data protection: Rules for the protection of personal data inside and outside the EU," [Online]. Available: https://commission.europa.eu/law/law-topic/data-protection_en

[42] "UAE enacts new federal personal data protection law," *Onetrust Blog*, Nov. 29, 2021. [Online]. Available: https://www.onetrust.com/blog/uae-enacts-new-federal-personal-data-protection-law

[43] "Understanding Saudi Arabia's Personal Data Protection Law (PDPL)," *Securiti.ai*, Mar. 19, 2025. [Online]. Available: https://securiti.ai/saudi-arabia-personal-data-protection-law/

[44] PwC, "Qatar Data Protection Law," *PwC Middle East*, [Online]. Available: https://www.pwc.com/m1/en/services/consulting/technology/cyber-security/navigating-data-privacy-regulations/qatar-data-protection-law.html

[45] A. Heraty, "Real estate tech strategies advance AI, workplace and sustainability," *JLL Newsroom* [Online]. Available: https://www.jll.com/en-us/newsroom/real-estate-tech-strategies-advance-ai-workplace-and-sustainability

[46] MIT News, "The real promise of synthetic data," *MIT News*, Oct. 16, 2020. [Online]. Available: https://news.mit.edu/2020/real-promise-synthetic-data-1016

[47] J. Near and D. Darais, "Differentially Private Synthetic Data," *NIST Cybersecurity Insights Blog*, May 03, 2021. [Online]. Available: https://www.nist.gov/blogs/cybersecurity-insights/differentially-private-synthetic-data

[48] S. Singh and F. Tekiner, "Exploring synthetic data generation with BigQuery DataFrames and LLMs," *Google Cloud Blog*, Jun. 12, 2024. [Online]. Available: https://cloud.google.com/blog/products/data-analytics/generate-synthetic-data-with-bigquery-dataframes-and-llms

[49] I. Goodfellow et al., "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

[50] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," *arXiv preprint arXiv:1611.07004*, Nov. 2016. [Online]. Available: https://arxiv.org/abs/1611.07004

[51] Z. Zhao, A. Kunar, R. Birke, and L. Y. Chen, "CTAB-GAN: Effective Tabular Data Generation for Mixed-Type Data with Conditional GAN," *arXiv preprint arXiv:2102.08369*, Feb. 2021. [Online]. Available: https://arxiv.org/abs/2102.08369

[52] L. Xu, K. Wenliang, M. Xiaochen, and L. Ying, "Modeling Tabular Data Using Conditional GAN," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. [Online]. Available: https://arxiv.org/abs/1907.00503

[53] J. Yoon, D. Jarrett, and M. van der Schaar, "Time-series Generative Adversarial Networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. [Online].

[54] Zillow Research, "Zillow Research," *Zillow.com*. [Online].
Available: https://www.zillow.com/research/
[55] Lianjia (Beike), "Lianjia (Beike)," *English.ke.com* [Online]. Available: https://english.ke.com/

[56] JSK Real Estate, "JSK Real Estate," [Online]. Available: http://jskre.com/.

[57] RealEstate Lebanon, "RealEstate Lebanon," [Online]. Available: http://realestate.com.lb/.

[58] BLOMINVEST Bank, "Real Estate Transactions – CAS," BRITE, [Online]. Available: https://brite.blominvestbank.com/category/Real-Estate-Transactions-CAS-3710

[59] The World Bank, Lebanon Economic Monitor, Spring 2021: Lebanon Sinking to the Top 3. [Online]. Available: https://www.worldbank.org/en/country/lebanon/publication/lebanon-economic-monitor-spring-2021-lebanon-sinking-to-the-top-3

[60] M. Dagher, "Lebanon's Economic Meltdown: A Deliberate Depression," Carnegie Middle East Center, Sep. 2020. [Online]. Available: https://carnegie-mec.org/2020/09/24/lebanon-s-economic-meltdown-deliberate-depression-pub-82782

[61] Reuters, "Lebanon's financial crisis and its ripple effects," Reuters, Aug. 12, 2021. [Online]. Available: https://www.reuters.com/world/middle-east/lebanons-financial-crisis-its-ripple-effects-2021-08-12/

[62] L'Orient Today, "In Lebanon, is real estate still a safe haven?," L'Orient Today, Dec. 2021. [Online]. Available: https://today.lorientlejour.com/article/1283623/in-lebanon-is-real-estate-still-a-safe-haven.html

[63] The National News, "How Lebanon's economic crisis has hit the property market," The National News, Sep. 2020. [Online]. Available: https://www.thenationalnews.com/business/property/how-lebanon-s-economic-crisis-has-hit-the-property-market-1.1070775

[64] S. P. Simchi-Levi and D. Simchi-Levi, "Getting Value from Monte Carlo Simulations," Harvard Business Review, Sep. 2014. [Online]. Available: https://hbr.org/2014/09/getting-value-from-monte-carlo-simulations

[65] Investopedia, "What Is a 'What-If' Analysis?," Investopedia. [Online]. Available: https://www.investopedia.com/terms/w/what-if-analysis.asp

[66] L. Rampini and F. Re Cecconi, "Artificial intelligence algorithms to predict Italian real estate market prices," *Journal of Property Investment & Finance*, Aug. 2021. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JPIF-08-2021-0073/full/html

[67] C. Kmen, G. Navratil, and I. Giannopoulos, "Location, Location, Location: The Power of Neighborhoods for Apartment Price Predictions Based on Transaction Data," *ISPRS International Journal of Geo-Information*, vol. 13, no. 12, p. 425, Dec. 2024. [Online]. Available: https://www.mdpi.com/2220-9964/13/12/425