Enhancing Real Estate Valuation through Deep Modeling of Spatiotemporal Dependencies and Feature Interactions

Lin Xu
Sichuan University
Chengdu, China
xulin12138@stu.scu.edu.cn

Yuankai Wu
Sichuan University
Chengdu, China
Kaimaogege@gmail.com

Abstract

Real estate valuation is crucial for both economic stability and societal development. Accurate property valuations are essential in guiding individual decisions and macroeconomic policies, especially in the context of rapid urbanization. Recent web platforms have transformed real estate markets by using data-driven models to predict property values. Despite these advancements, challenges remain in modeling the complex interactions between continuous and categorical factors and in addressing the spatio-temporal dependencies inherent in real estate transactions. To address these challenges, we propose **TabularCNP** (Tabular Conditional Neural Process), a novel framework that leverages both deep learning models on tabular data and Conditional Neural Processes (CNPs). The framework incorporates deep tabular models to handle complex feature interactions, while conditioning the target property's value on nearby recent sales to capture spatiotemporal dependencies. By transforming real estate valuation into a tabular data-driven machine learning task, we effectively consider the intricate interactions between features. Additionally, we introduce a CNP-based event logging module to capture the influence of nearby transactions. Extensive experiments on real-world datasets validate the proposed framework, demonstrating significant improvements in prediction accuracy by effectively modeling multifactor interactions and spatiotemporal dynamics. The code is available at https://anonymous.4open.science/r/TabularCNP-EE04/.

CCS Concepts

• **Computing methodologies** → **Machine learning**; • **Information systems** → **Data mining**; • **Applied computing** → **Marketing**.

Keywords

House Price Prediction, Conditional Neural Networks, Deep Learning, Spatiotemporal Dependencies, Recommendation Systems

ACM Reference Format:
Lin Xu and Yuankai Wu. 2025. Enhancing Real Estate Valuation through Deep Modeling of Spatiotemporal Dependencies and Feature Interactions. In *Companion Proceedings of the ACM Web Conference 2025 (WWW Companion '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3701716.3717367

WWW Companion '25, April 28-May 2, 2025, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1331-6/2025/04
https://doi.org/10.1145/3701716.3717367

1 Introduction

Real estate valuation plays a critical role in both the global economy and societal development. Accurate property valuations are fundamental to the stable functioning of real estate markets, providing both buyers and sellers with fair pricing benchmarks [9] and mitigating the risk of market imbalances due to price volatility. Furthermore, precise property assessments are crucial to the financial sector, where banks and financial institutions rely on them to issue mortgage loans and manage risk, thus contributing to overall financial stability [1]. Governments also use property valuations to determine real estate taxes, ensuring fairness and transparency in taxation [26]. As urbanization accelerates, accurate property valuation not only guides individual decisions but also supports macroeconomic regulation and policy-making. This helps allocate social resources efficiently and promotes sustainable economic growth. In recent years, web applications such as China's Beike and the U.S.-based Zillow have revolutionized the real estate industry by offering real-time, data-driven property valuations [14]. These platforms collect vast amounts of user data and property-related information, along with transparent pricing details, making it possible to leverage data-driven machine learning models for real estate valuation.

Web platforms have made significant strides in utilizing vast datasets, yet the complexity of the real estate market requires more than just access to data. Predicting property values involves not only understanding traditional physical and socioeconomic variables but also addressing more intricate challenges. Figure 1 illustrates the various factors that need to be considered in the real estate valuation process. These factors include property features such as listing price, building area, number of rooms, and the availability of amenities like elevators, all of which interact in complex ways to determine the final property value. Moreover, the value of the target property is influenced by the past transactions of nearby properties. In summary, the real estate valuation problem presents two main challenges.

Challenge 1 Multifaceted Influences and Interactions on Valuation: The task of accurately predicting real estate value is complicated by the need to model a diverse array of influencing factors. The price determination process is influenced by continuous variables, such as listing prices, the sizes of properties, average prices within a specific area, the convenience of nearby amenities, and the age of the properties. Simultaneously, a range of categorical variables—such as the availability of elevators and the reputation of the managing property company—also exert significant influence. This complexity

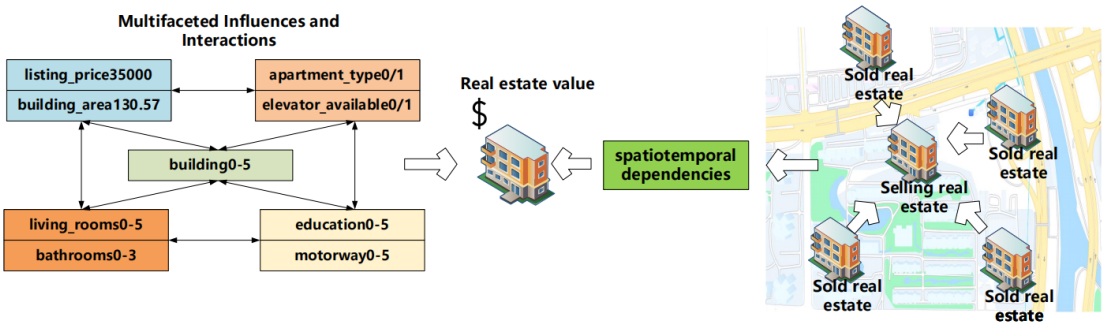

Figure 1: Illustration of the multifaceted influences and interactions on real estate valuation. Additionally, spatiotemporal dependencies from nearby sold properties further influence the valuation process.

is compounded by the challenge of modeling the intricate interactions among these variables, each of which contributes in a unique way to shaping the final market price of a property. Successfully addressing this challenge requires sophisticated modeling techniques that can capture and quantify the complex, dynamic interplay of these multifarious factors.

Challenge 2 Spatiotemporal Dependencies: Predicting real estate prices introduces the nuanced challenge of accounting for spatiotemporal dependencies, a reflection of the deeply social nature of real estate transactions. Properties are influenced not merely by static features but dynamically by the recent sale prices of nearby properties, a factor increasingly pertinent in the digital milieu of online trading platforms. This necessitates viewing each property not just as an isolated entity but as part of a broader, interconnected spatiotemporal process [29]. The proximity of properties, both in time and space, can lead to pronounced mutual influences, echoing through the market like ripples. Thus, the challenge lies in developing predictive models capable of integrating these temporal and spatial dynamics to accurately forecast house prices in a market characterized by constant flux.

To address these challenges, we propose a novel framework, **TabularCNP** (Tabular Conditional Neural Process), that tackles both issues. We draw inspiration from machine learning for tabular data, which, like real estate data, contains numerous continuous and categorical variables. Deep tabular Models [11, 25] have significantly improved prediction accuracy by capturing both low-order and high-order feature interactions. Rather than treating each property solely as tabular data, we condition the price on nearby sales data to capture the spatiotemporal relationship [2]. Our key assumption is that the price of a new property is conditioned on the prices of nearby properties recently sold, aligning with socioeconomic assumptions that local market prices are influenced by neighboring property sales [37]. We feed historical transaction data from surrounding properties into a neural network encoder, generating a representation that reflects the influence of these neighboring properties. This representation, combined with the target property's own attributes, is then fed into models like WDL, as with traditional tabular data, to produce the final value prediction. The main contributions of this paper include:

- Our first key contribution is the introduction of the TabularCNP framework, which effectively CNP with advanced tabular data models. This integration allows for a more nuanced and dynamic approach to real estate valuation by not only capturing complex interactions between continuous and categorical factors, but also incorporating the spatiotemporal dependencies of nearby properties.
- We propose an event logging module based on a Conditional Neural Process (CNP) to identify and log transactions of nearby properties within spatiotemporal proximity. This approach effectively models the influence of surrounding properties and their historical transaction prices on the current property's value.
- We conduct extensive experiments on multiple real-world datasets to validate the accuracy and effectiveness of the proposed framework. Additionally, experiments clearly demonstrate that leveraging the principles of the CNP to model spatiotemporal dependencies significantly improves prediction accuracy.

2 Related Works

2.1 Real Estate Valuation

Real estate valuation has been a longstanding research focus due to its economic significance. Traditional models, such as Hedonic Price Models, use linear regression to assess how property features affect house prices [37], while more advanced methods, like geospatial econometrics, address spatial dependencies but struggle with regional heterogeneity [2]. With the rise of machine learning, models such as Support Vector Machines (SVM) [13] and tree-based methods like Random Forests [5] have improved prediction accuracy by handling complex interactions [43] and integrating diverse data types, including geographic, socioeconomic, and temporal features. Modern approaches now incorporate points of interest (POI) and mobility patterns to better capture the influence of location and human activity on property values [31]. However, these works often ignore the interactions between different features. Our work not only takes into account various types of features but also leverages deep neural networks to model their interactions [32].

Another key factor is the spatiotemporal dependencies in real estate value fluctuations. Recent research shows that property values are influenced by nearby properties and historical transactions. The use of graph neural networks (GNNs) improves the modeling efficiency of these dependencies. Models such as MugRep use graph neural networks to capture asynchronous spatiotemporal dependencies , improving the prediction effect of real estate valuation .[41].

2.2 Deep Learning for Tabular Data

Deep learning has demonstrated significant potential for processing tabular data, which is widely used in healthcare, finance, and transportation [12, 25]. Early approaches, such as fully connected networks (FCNs), struggled to model complex feature interactions. To address this, TabNet [4] introduced sequential attention for instance-wise feature selection, enhancing interpretability and performance. SAINT [38] further improved scalability by combining self-attention with inter-sample attention, while TabTransformer [30] integrated attention mechanisms with MLPs to process both categorical and numerical data. The FT-Transformer [22] refined transformer architectures for tabular datasets, optimizing efficiency. More recently, GNN4TDL and GANDALF [38] have incorporated graph-based techniques, bridging neural networks and decision trees to better model feature relationships. These developments highlight ongoing efforts to enhance deep learning performance for tabular data while maintaining computational efficiency.

2.3 Conditional Neural Processes

Conditional Neural Processes (CNPs) [18, 19] are a family of meta-learning models designed to map observed data points to predictive stochastic processes [15]. By conditioning on an observed context set for each target point, CNPs can effectively model tasks that involve capturing local dependencies, making them particularly suitable for real estate valuation where geographic and socioeconomic features play critical roles. CNPs have been shown to generalize well to unseen tasks by leveraging the shared structure between tasks and incorporating uncertainty into their predictions, which is crucial for tasks with sparse data, such as property valuation in regions with very low samping rate. Several extensions to the original CNP framework have been proposed to improve both the flexibility and expressiveness of the model [7, 17, 21, 28, 36, 40, 42].

3 Problem formulation

In this section, we introduce several key definitions and formally define the real estate valuation problem.

Definition 3.1. **Real Estate Dataset**: Consider a real-world dataset $\mathcal{S} = \{(\mathbf{x}_{t,s}, y_{t,s})\}$, where $\mathbf{x}_{t,s} \in \mathbb{R}^n$ represents the n-dimensional feature vector of the property at time t and location s (e.g., building area, district name, etc.), and $y_{t,s}$ denotes the price per square meter.

The features $\mathbf{x}_{t,s}$ in our dataset are categorized into five primary groups, each addressing different aspects crucial to property valuation.

(1) **Real Estate Profile** features encompass physical attributes of the property, such as total floors, building area, number of rooms, as well as market engagement metrics (e.g., viewing count and follower count).
(2) **Geographical** features represent spatial information, such as geographical coordinates and neighborhood location, which reflect the property's location and its proximity to important commercial areas. In addition to the geographical information of the target property, we also consider Points of Interest (POI) data, which include functional facilities, roadways and economic activities.
(3) **Building Attributes** describe the type, structure, orientation, heating method, and whether the property is equipped with elevators.
(4) **Transaction and Usage** features include detailed information on the property's transaction rights, intended usage, and ownership status, which are critical for predicting the property's market performance and investment potential.
(5) **Listing Information** captures time-based data such as listing date, providing insights into the market dynamics over time and their influence on property prices.

The problem we aim to address is, when a user uploads a second-hand property to a web platform, we can provide an accurate valuation for the property by leveraging the dataset $\mathcal{S}$

Definition 3.2. **Real Estate Valuation**: Given a target property $(\mathbf{x}_{i,j}, y_{i,j})$ from the dataset $\mathcal{S}$, the task of **Real Estate Valuation** involves predicting the transaction price $y_{i,j}$ of the target property using both its own feature vector $\mathbf{x}_{i,j}$ and an observation set $O_{i,j} = \{(\mathbf{x}_{t',s'}, y_{t',s'}) \mid i-w < t' < i-1, s' \in \mathcal{N}_j\}$, where w is a predefined time window that selects the most recent transactions prior to the target property, and $\mathcal{N}_j$ is the neighborhood set for location j.

4 Methodology

In the following sections, we present the background and architecture of our proposed model. As shown in Figure 2, the framework consists of three main modules: the **Condition Set Extraction**, the **Encoder**, and the **Decoder**.The **Condition Set Extraction** module is responsible for extracting $O_{i,j}$ from $\mathcal{S}$. The **Encoder** is a neural network that captures the information from the condition set, while the **Decoder** is used to make the final prediction.

In more detail, we use the following architecture:

$$\begin{aligned} \mathbf{r}_{t',s'} &= h_{\theta_\mathbf{e}}(\mathbf{x}_{t',s'}, y_{t',s'}) \quad \forall (\mathbf{x}_{t',s'}, y_{t',s'}) \in O_{i,j}, \\ \mathbf{r}_{i,j} &= \rho\left(r_{i-w,1}, r_{i-w+1,1}, \cdots, r_{i-1,n_j}\right), \\ \phi_{i,j} &= g_{\theta_\mathbf{d}}(\mathbf{x}_{i,j}, \mathbf{r}_{i,j}), \end{aligned} \tag{1}$$

where $h_{\theta_\mathbf{e}}$ and $g_{\theta_\mathbf{d}}$ are the encoder and decoder neural networks, respectively. $\theta_\mathbf{e}$ and $\theta_\mathbf{d}$ represent the parameters for the encoder and decoder. The aggregation function ρ is permutation-invariant and input-size agnostic. Specifically, we evaluate three types of ρ: MEAN [19], DeepSets [44], and Attention [17]. Detailed explanations of these methods are provided in Appendix B. In the real estate valuation setting, given several examples of $\forall(\mathbf{x}_{t',s'}, y_{t',s'}) \in O_{i,j}$, the task is to output $\mathbf{r}_{i,j}$ while ensuring permutation invariance with respect to the predictors (i.e., the result is independent of the input order of elements in $O_{i,j}$). In traditional CNP models [20], $\phi_{i,j}$ typically represents a distribution. However, in this work, when comparing with traditional tree-based methods [6], we output $\phi_{i,j}$

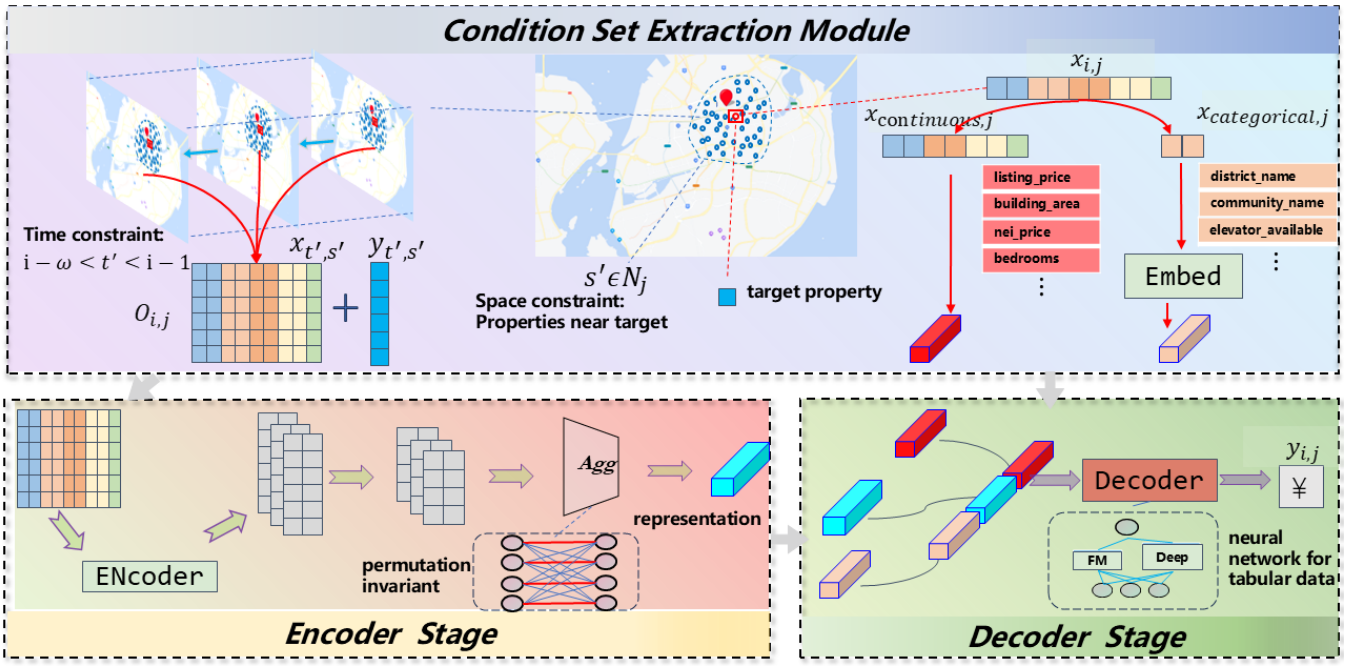

Figure 2: Model framework of TabularCNP: The model consists of three components: the condition set extraction module, the encoder, and the decoder. The condition set is selected based on spatiotemporal constraints to identify historical transactions related to the target property. The encoder captures representations of these related transactions, while the decoder uses deep learning on tabular data to predict the property's price.

as a single fixed value. This is equivalent to treating the real estate value as a Gaussian distribution with zero variance. We can also set $\phi_{i,j}$ as the parameters of a distribution, which would allow us to quantify the uncertainty in the real estate value.

4.1 Condition Set Extraction Module

In the original CNP, the condition (context) set is randomly selected from all observed points [20]. However, in our problem, as mentioned earlier, the value of a property is only influenced by nearby properties and recent transactions. It's hard to imagine that the price of a house sold 10 years ago in Beijing would impact the price of a property currently for sale in a third-tier city in Southwest China. Therefore, when constructing the condition set for each property, we use a time window and spatial range search method to avoid conditioning on unnecessary information. The process can be summarized as follows:

Given the feature vector of the target property $(\mathbf{x}_{i,j})$ and the dataset $\mathcal{S}$ containing all historical transaction records, we begin by initializing an empty condition set $O_{i,j}$. The start of the time window is defined as $i - w$, where w represents the predefined time window. We then filter the historical transactions in $\mathcal{S}$ that occurred within the time range $[i - w, i - 1]$. Once these relevant historical transactions are identified, we apply a second filter based on spatial proximity. Specifically, we check if the Euclidean distance between the target property and each transaction property $(\mathbf{x}_{t',s'}, y_{t',s'})$ is less than the maximum spatial search distance L. For each transaction that meets both the temporal and spatial criteria, the transaction $(\mathbf{x}_{t',s'}, y_{t',s'})$ is added to the condition set $O_{i,j}$. The final condition set is then returned.

4.2 Tabular Neural Networks for Encoder/Decoder

In the proposed TabCNP framework, the encoder-decoder architecture synergistically operates to extract and leverage critical information from the input conditional set $O_{i,j}$, thereby generating the final predictive output. The encoder network, h_{θ_e}, is tasked with transforming the input feature set into a high-dimensional representation that encapsulates all the essential information required for precise predictions. Specifically, the encoder processes each pair $(\mathbf{x}_{t',s'}, y_{t',s'})$ from the conditional set, mapping it to a corresponding condition vector $\mathbf{r}_{t',s'}$. These condition vectors are subsequently aggregated via the function ρ, yielding a final condition vector $\mathbf{r}_{i,j}$ that embodies the aggregate interactions of the input features. The mathematical formulation for this aggregation is given by:

$$\mathbf{r}_{i,j} = \rho\left(f_{\text{tabular encoder}}(\mathbf{x}_i, y_i)\right). \tag{2}$$

The decoder network, $f_{\text{tabular decoder}}$, then utilizes the encoded condition vector $\mathbf{r}_{i,j}$ in conjunction with the input feature $\mathbf{x}_{i,j}$ to produce the final predicted value $\phi_{i,j}$, as expressed by:

$$\phi_{i,j} = f_{\text{tabular decoder}}(\mathbf{x}_{i,j}, \mathbf{r}_{i,j}). \tag{3}$$

Distinct from the original Conditional Neural Processes (CNP) framework, which deploys a Multi-Layer Perceptron (MLP) for the

decoding process, the TabularCNP framework introduces deep tabular models into the decoder component. This significantly amplifies the model's capacity to capture intricate feature interactions. The encoder-decoder architecture enables the system to capture complex feature interactions while also incorporating domain-specific knowledge. Specifically, the encoder extracts essential information from the conditional set, thereby enhancing the decoder's ability to model the complex relationships inherent in the data.

4.3 Training TabularCNPs

As previously mentioned, the output of TabularCNPs can take various forms, but we can summarize its loss function as minimizing the negative conditional log probability of the training data:

$$\mathcal{L}(\theta) = -\mathbb{E}_{f\sim P}\left[\mathbb{E}_N\left[\log Q_\theta\left(y_{i,j} \mid O_{i,j}, \mathbf{x}_{i,j}\right)\right]\right], \tag{4}$$

where $y_{i,j}$ represents the true transaction price, and $\mathbf{x}_{i,j}$ is the input feature vector of the property. This negative log likelihood objective function is standard for models that generate probability distributions as output.

When the model outputs a single value and we assume that real estate values follow a Gaussian distribution with zero standard deviation, this becomes equivalent to minimizing the Mean Squared Error (MSE) loss:

$$\mathcal{L}_{\text{MSE}}(\theta) = \text{Mean}\left(\left(y_{i,j} - \hat{y}_{i,j}\right)^2\right), \tag{5}$$

where $y_{i,j}$ is the true transaction price, and $\hat{y}_{i,j}$ is the predicted value. This formulation simplifies the task to a regression problem with Gaussian assumptions, where the MSE loss captures the squared difference between predicted and true values.

4.4 Autoregressive Inference

In our Condition Set Extraction Module, we only considered information from sold real estate properties. However, currently listed properties may also have relationships with other listed properties. To address this issue, we can adopt the idea of autoregressive CNPs [8]. Without any modifications to the model or training procedure, the autoregressive approach can be applied to solve this problem.

Specifically, we can use the trained model to predict the values of nearby listed properties around the target property.

$$\hat{y}_{s''} = f(\mathbf{x}_{s''}), \quad \text{where } s'' \in \mathcal{N}_j. \tag{6}$$

Next, we add the set $\hat{O}_j$, consisting of features $\mathbf{x}_{s''}$ and predicted values $\hat{y}_{s''}$, with $s'' \in \mathcal{N}_j$ to the context set $O_{i,j}$, resulting in $O_{i,j} \cup \hat{O}_j$. The constructed model is then used to predict the target property's value.

5 Experiments

5.1 Experimental Setup

5.1.1 Datasets. We conduct experiments on three real-world second-hand housing transaction datasets, covering three major cities: Shanghai, Nanjing, and Xiamen. The datasets were chronologically sorted by transaction dates. For the Shanghai dataset, the data spans from June 2023 to June 2024, containing 31495 transactions. For the Xiamen dataset, we use transaction data from January 2015 to June 2024, containing 28394 transactions. For Nanjing, the dataset includes data from January 2022 to March 2024, with 45104 transactions. Each dataset was split into training, validation, and testing sets following a 64%/16%/20% ratio, based on the chronological order of transaction dates. Our prediction target is the transaction prices of these houses. To normalize the impact of house area, the target we predict is the price per square meter in RMB. Due to the ongoing housing market downturn in China, we observe that property prices in the test set are lower than those in the training set. For each property, we used a total of 61 continuous features and 14 categorical features.

5.1.2 Implementation Details. For the Observation Set Extraction module in our framework, we set the time window w = 3month (i.e., properties sold within 3 months). Properties within a spatial distance of 1.1 km from the target property are considered part of the condition set. We evaluated three methods as decoders: DeepFM [25], WDL [12], and TabNet [4]. For the encoder, we retained the MLP originally used in the CNP, as it provides a strong baseline for tabular data representation. The permutation invariant function used to aggregate representations of each property in the condition set was evaluated with three approaches: simple mean, attention [39], and deep sets [44].

When comparing with other models, we used the basic Mean aggregation function for the representation of the condition set and did not employ autoregressive inference. These two aspects will be further discussed in the ablation studies later in the paper. All deep learning models were trained using the Adam optimizer with an initial learning rate of 0.01. A learning rate decay strategy was applied with a step size of 10 and a decay factor $\gamma = 0.1$. Early stopping was employed, halting training if the validation loss did not improve for more than 5 consecutive epochs.

5.1.3 Evaluation Metrics. For comparison with methods like GBRT, we used point estimation for output, assuming that real estate property values follow a Gaussian distribution with zero variance. We adopt three widely used metrics: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE). All predicted and actual transaction prices are reported in Chinese Yuan (CNY). The metrics are defined as follows:

$$\begin{aligned}
\text{MAE} &= \frac{1}{|\mathcal{S}_{\text{te}}|}\sum_{i=1}^{|\mathcal{S}_{\text{te}}|}|\hat{y}_i - y_i|,\\
\text{MAPE} &= \frac{1}{|\mathcal{S}_{\text{te}}|}\sum_{i=1}^{|\mathcal{S}_{\text{te}}|}\left|\frac{\hat{y}_i - y_i}{y_i}\right|,\\
\text{RMSE} &= \sqrt{\frac{1}{|\mathcal{S}_{\text{te}}|}\sum_{i=1}^{|\mathcal{S}_{\text{te}}|}(\hat{y}_i - y_i)^2}.
\end{aligned} \tag{7}$$

where $\hat{y}_i$ is the predicted value, y_i is the actual value, and $|\mathcal{S}_{\text{te}}|$ is the total number of samples in the test set $\mathcal{S}_{\text{te}}$.

5.1.4 Baselines. We evaluate our proposed model against eight well-established baselines: Gradient Boosting Regression (GBR) [16], eXtreme Gradient Boosting (XGB) [10], Random Forest (RF) [27], Linear Regression (LR), multilayer perceptron (MLP), Deep Factorization Machine (DeepFM) [25], Width & Deep Learning (WDL) [12], and TabNet [4].

Table 1: Performance comparison across models for three cities: Xiameng, Nanjing, and Shanghai. The best results are highlighted in green, second-best in yellow, and worst in red. Models marked with * are our proposed models.

Model	Xiamen			Nanjing			Shanghai		
	MAE	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**
Baseline models									
GBR	2330.7762	3448.9986	6.68%	1760.2173	2442.0121	8.58%	3129.1376	4551.2679	6.96%
XGB	2490.0595	3839.78	7.00%	1295.1652	1962.3659	5.75%	3120.0101	4527.5928	6.42%
RF	3547.3348	5672.0390	9.63%	1186.0382	1890.1769	5.14%	2842.6732	4124.8215	6.37%
LR	7061.9934	9222.0543	22.95%	3455.1958	5038.5065	17.69%	6471.6468	10920.3349	15.48%
MLP	1992.7139	3154.5278	6.08%	1446.4807	2132.8599	7.5%	3026.2814	4935.5771	7.02%
DeepFM	2362.9547	3559.6692	7.00%	1326.7294	2052.704	6.14%	2918.9079	4804.4914	6.68%
WDL	2315.982	3580.6795	6.76%	1283.1952	1923.3265	6.19%	2868.1252	4742.7284	6.34%
TabNet	3784.7710	4926.7070	13.21%	1645.8720	2181.925	9.15%	2826.4720	4020.2790	7.44%
TabularCNP models									
WDL*	2161.4643	3319.9164	5.46%	1284.3615	1979.1063	4.99%	2766.8457	4466.2539	5.38%
DeepFM*	1907.2057	2921.8227	4.77%	1165.814	1831.7726	4.50%	2861.1035	4549.2441	5.59%
TabNet*	3520.7004	4723.2230	12.19%	1540.6690	2087.1730	8.47%	2655.2480	3679.4200	6.32%

5.2 Performance Evaluation

Table 1 presents the overall results of all methods, alongside comparisons with baseline models across the three datasets and three evaluation metrics. The results with the conditional neural process (CNP) are shown at the bottom of the table, with models represented in the Encoder+Decoder format. Key observations are as follows: 1) Clearly, *transforming the real estate valuation problem into a machine learning task on tabular data is effective*, as evidenced by the superior performance of WDL, DeepFM and TabNet compared to MLP. The MLP model lacks the ability to capture feature interactions and does not have specialized treatment for categorical and continuous variables, resulting in worse performance in our experiments. Even with only the basic deep learning models for tabular data, like WDL and DeepFM, we observed significantly better results than MLP. Moreover, while the debate between deep learning and tree-based methods for tabular data continues [23, 34], our findings suggest that deep learning methods perform better for the real estate valuation problem. 2) After introducing CNP, the property price is not only influenced by its own features but also by the historical sales data of surrounding properties. We observed that *models incorporating CNP consistently outperformed those without it across all three datasets*. Additionally, we found that even when using MLP as the encoder, the model still performed very well. This suggests that when generating a representation in the encoder to capture spatiotemporal dependencies, modeling feature interactions is not particularly important.

5.3 Feature Importance Analysis

In this section, we analyze the feature importance of our proposed model using SHAP (SHapley Additive exPlanations) values [33], a widely adopted method for interpreting machine learning models. We conduct the evaluation on **DeepFM***, as it achieves the best performance in Table 1. Figure 3 illustrates the top 15 most significant features across our datasets, ranked by their mean absolute SHAP values.

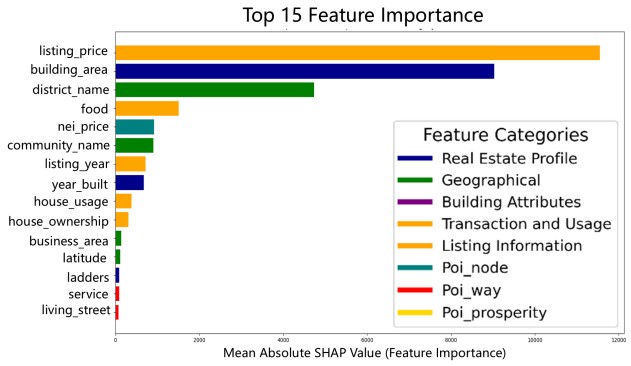

Figure 3: Top 15 Feature Importance with Category Colors. The most important features across all datasets, ranked by their mean absolute SHAP value.

From the figure, it is clear that **listing_price**, **building_area**, and **district_name** hold the greatest importance in determining house prices. This aligns with domain knowledge in real estate, where **listing_price** plays a dominant role, indicating that the initial price listed significantly impacts both the perceived and actual value of a property. A particularly noteworthy feature is **nei_price**, a manually constructed feature that represents the average price of nearby properties sold in the past. This feature captures the influence of historical sales within the vicinity of the target property. The importance of **nei_price** in our model underscores the strong spatiotemporal dependencies inherent in property valuation, reflecting how buyers and sellers often adjust their expectations based on recent transactions in the same neighborhood. Additionally, Point of Interest (POI) features such as **food** and **service** are

also notable, highlighting the importance of nearby amenities. Interestingly, food-related POI information appears to have a strong correlation with property prices. This may be because areas with a high concentration of restaurants are often located near central business districts (CBDs) and downtown areas, where property prices tend to be higher. Temporal features like **listing_year** and **year_built** contribute significantly, emphasizing the relevance of a property's age and the timing of its market entry. Lastly, geographical features such as **latitude** and **community_name** underscore the crucial role of spatial location in property valuation, as properties in more desirable locations tend to command higher prices.

5.4 Ablation Study

We conduct the following ablation studies to verify the impact of each component in our model. Similar to the feature importance analysis, we use the **DeepFM*** model for this study.

5.4.1 Effect of autoregressive inference. In Table 2, ARCNP refers to the inclusion of predicted prices for nearby properties currently on the market in the condition set for the target property, while CNP means these properties are not included. As observed in Table 2, the autoregressive inference strategy (ARCNP) demonstrates marginal improvements over the standard inference strategy (CNP) across all cities. For example, in Nanjing, the ARCNP model achieves a lower RMSE (1779.13) compared to CNP (1831.77), with corresponding decreases in MAE and MAPE. Similarly, in Shanghai, autoregressive inference results in a slight reduction in RMSE and MAPE, although the differences are less pronounced. This improvement can be attributed to the autoregressive inference's ability to leverage additional spatiotemporal dependencies that may not be fully captured by the standard inference method. However, it is important to note that using the autoregressive approach increases the number of inferences required, which raises the computational complexity during the inference process.

5.4.2 Effect of aggregation function. In this experiment, we aim to investigate how different aggregation methods for the condition set representation impact the model's predictive performance. The aggregation function plays a crucial role in condensing information from multiple observation points into a single representation, which is then used for target property value prediction. Table 3 summarizes the results of this ablation study. Overall, the attention-based aggregation function achieved the best performance across the three cities, as it can automatically learn the contribution of different properties in the condition set to the target price. DeepSets performed slightly better than the mean-based method, as it applies a nonlinear transformation to the sum of all property representations using a neural network. The mean-based aggregation had the weakest performance but was the most computationally efficient due to its simplicity.

5.4.3 Effect of feature selection. To further analyze the influence of different features on our model, we conducted a feature ablation study. We selected the top three most important features based on SHAP analysis: *listing_price*, *building_area*, and *district_name*. The feature ablation experiment involved removing one of these features and training the model without it. The versions are denoted as noLp (no *listing_price*), noBa (no *building_area*), and noDn (no *district_name*). The performance was compared to the full-featured version of **DeepFM*** (complete). The results in Figure 4 clearly show that removing any of the three key features degrades performance, as indicated by the higher RMSE values. Among them, *listing_price* proved to be the most critical feature, as its absence led to the largest performance drop across all datasets, particularly in Xiamen and Nanjing. The listing price reflects the seller's psychological expectation of the property's value and typically serves as the starting point for negotiations between the buyer and seller. Removing the building area feature can also negatively impact the accuracy of house price predictions. In real estate, when buyers compare two properties in the same location with similar characteristics, building area becomes a critical factor in differentiating their value. A property with a larger building area provides more usable space, which significantly enhances its perceived value compared to a smaller property.

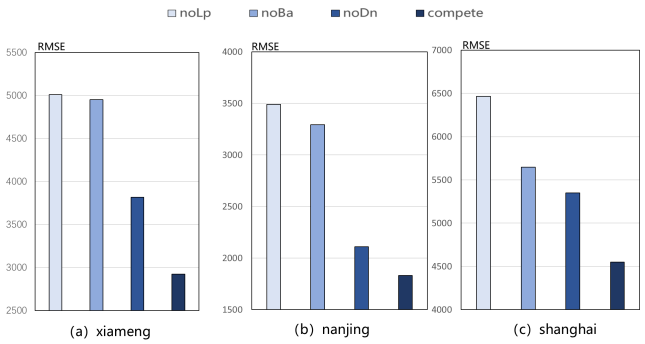

Figure 4: Effect of Feature Selection.

5.4.4 Effect of Spatial Proximity Thresholds for the Condition Set. In this section, we present a distance ablation study to investigate the impact of different spatial proximity thresholds on the performance of our model. Given that each property is geographically positioned by its latitude and longitude, we conducted experiments by varying the Euclidean distance thresholds within the geographic coordinate space to define the observation set. Specifically, we set three different distance thresholds: 500 meters, 1 kilometer, and 1.5 kilometers in real-world terms. The rationale behind this experiment is to evaluate the model's sensitivity to the spatial proximity of nearby transactions when constructing the condition set. For each distance threshold, we keep the temporal constraints constant, ensuring that only the spatial distance varied. Figure 5 shows the results of this study across three cities: Xiamen, Nanjing, and Shanghai.

In Xiamen and Shanghai, the model performs best when the spatial threshold is set to 1 kilometer, but its performance deteriorates when expanded to 1.5 kilometers, indicating that including more distant transactions does not provide additional benefits. In contrast, in Nanjing, a 1.5-kilometer threshold yields better results, possibly due to a broader spatial correlation in its real estate market. Whereas in Xiamen and Shanghai, where land supply is more constrained, markets tend to be more localized, and property prices are primarily influenced by nearby transactions. This aligns with the local market effect and spatial heterogeneity theory [3], which

Table 2: Performance comparison between ARCNP and CNP across models for three cities: Xiamen, Nanjing, and Shanghai.

Model	Xiamen			Nanjing			Shanghai		
	MAE	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**
CNP	**1907.2057**	2921.8227	4.77%	1165.8140	1831.7726	4.50%	2861.1035	4549.2441	5.59%
ARCNP	1909.8864	**2908.8138**	**4.72%**	**1124.1621**	**1779.1323**	**4.44%**	**2856.8240**	**4531.4805**	**5.44%**

Table 3: Performance comparison between Attention, DeepSets, and MEAN aggregation across methods for three cities: Xiamen, Nanjing, and Shanghai.

Method	Xiamen			Nanjing			Shanghai		
	MAE	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**
Attention	**1906.1232**	**2913.8404**	4.88%	**1138.2456**	**1632.7159**	**4.48%**	**2727.3826**	**4150.7861**	**5.48%**
DeepSet	1906.2169	2923.5541	**4.75%**	1158.1879	1776.3424	4.51%	2827.7438	4553.2322	5.51%
Mean	1907.2057	2921.8227	4.77%	1165.814	1831.7726	4.50%	2861.1035	4549.2441	5.59%

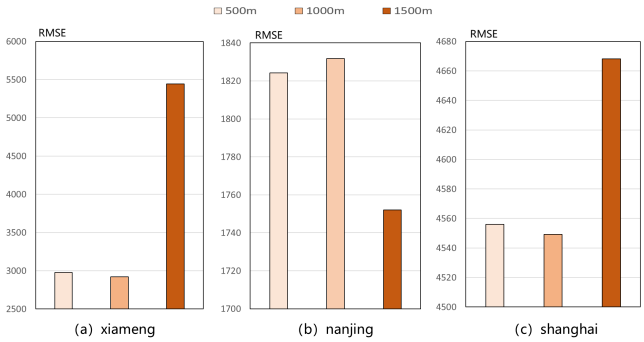

Figure 5: Effect of Distance Thresholds for the Condition Set.

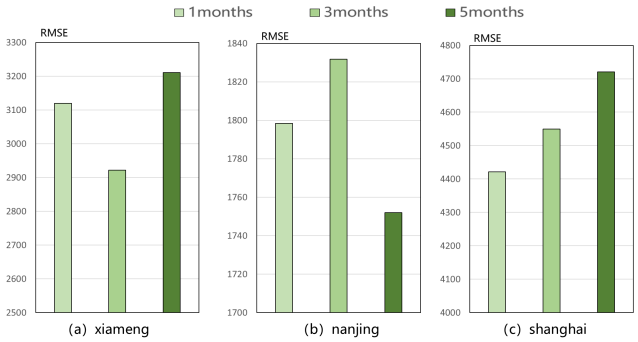

Figure 6: Effect of Time Window Thresholds for the Condition Set.

suggests that in areas with concentrated demand and limited supply, property prices exhibit stronger correlations within smaller geographic regions.

5.4.5 Effect of Time Window Thresholds for the Condition Set. In this section, we conduct a time window ablation study to explore the impact of varying temporal proximity on the model's performance. Specifically, we examine how different time windows for selecting the condition set influence the predictive accuracy of our model. We experiment with three distinct time windows: 1 month, 3 months, and 5 months. These windows define the temporal constraints for selecting relevant historical transactions that are used to predict the price of a target property, with the spatial distance kept constant across all experiments. The goal is to understand the sensitivity of our model to the age of the historical data used in predictions.

This result indicates that the optimal time window varies across cities. Shanghai, as the most economically developed city with the tightest land supply, has a fast information flow, aligning with the **Efficient Market Hypothesis (EMH)** [35], where a 1-month window is sufficient to capture market trends, making historical data less influential. In contrast, Xiamen (optimal at 3 months) and Nanjing (longer window preferred) exhibit slower market responses and **Information Lag** [24], causing delayed price adjustments, thus making longer transaction histories more valuable for prediction.

6 Conclusion

In this work, we proposed **TabularCNP**, a novel framework for real estate valuation that leverages deep learning models on tabular data and conditional neural processes to address the challenges of modeling multifactor interactions and spatiotemporal dependencies. Our approach incorporates feature interaction models to effectively capture complex feature relationships, while the introduction of a Conditional Neural Process (CNP)–based event logging module allows us to condition the target property's value on nearby recent sales, capturing spatiotemporal dynamics. Extensive experiments on real-world datasets demonstrated that **TabularCNP** significantly outperforms baseline models in terms of prediction accuracy, with performance gains particularly notable in capturing the spatial and temporal nuances of real estate transactions. For future work, we aim to explore the inclusion of additional factors, such as economic indicators and urban planning data, to further enhance the model's predictive capabilities.

References

[1] Tobias Adrian and Hyun Song Shin. Liquidity and financial cycles. Technical report, BIS Working Papers, 2008.

[2] Luc Anselin. Local indicators of spatial association—lisa. *Geographical analysis*, 27(2):93–115, 1995.

[3] Luc Anselin. *Spatial econometrics: methods and models*, volume 4. Springer Science & Business Media, 2013.

[4] S. Ö Arik and T. Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

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

[6] Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone. *Classification and regression trees*. Wadsworth & Brooks/Cole, 1984.

[7] Thomas Bruinsma et al. Adapting conditional neural processes for time-series prediction. *Journal of Machine Learning Research (JMLR)*, 2023.

[8] Wessel P Bruinsma, Stratis Markou, James Requiema, Andrew YK Foong, Tom R Andersson, Anna Vaughan, Anthony Buonomo, J Scott Hosking, and Richard E Turner. Autoregressive conditional neural processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[9] Karl E. Case and Robert J. Shiller. Is there a bubble in the housing market? *Brookings Papers on Economic Activity*, 2003(2):299–362, 2003.

[10] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[11] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Harsha Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10, 2016.

[12] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Harsha Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10, 2016.

[13] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[14] Liran Einav and Jonathan Levin. Economics in the age of big data. *Science*, 346(6210):1243089, 2014.

[15] Andrew Foong et al. Meta-learning for stochastic processes. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2020.

[16] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[17] Marta Garnelo et al. Attentive neural processes. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2018.

[18] Marta Garnelo et al. Neural processes. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2018.

[19] Marta Garnelo, Dan Rosenbaum, et al. Conditional neural processes. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.

[20] Marta Garnelo, Dan Rosenbaum, et al. Conditional neural processes. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1704–1713, 2018.

[21] Jonathan Gordon et al. Robust neural processes for noisy datasets. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2019.

[22] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[23] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520, 2022.

[24] Sanford J Grossman and Joseph E Stiglitz. On the impossibility of informationally efficient markets. *The American economic review*, 70(3):393–408, 1980.

[25] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: A factorization-machine based neural network for ctr prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1725–1731, 2017.

[26] Joseph Gyourko and Raven Molloy. *Regulation and housing supply*, pages 1289–1337. Elsevier, 2015.

[27] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[28] Thomas Holderrieth et al. Extending cnps for noisy real-world tasks. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[29] Bo Huang, Bo Wu, and Michael Barry. Geographically and temporally weighted regression for modeling spatio-temporal variation in house prices. *International journal of geographical information science*, 24(3):383–401, 2010.

[30] Xiangrui Huang, Dan Yao, and Xinyuan Zhou. Tabtransformer: A transformer model for tabular data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[31] C. Kang, X. Ma, and D. Tong. Real estate price prediction using poi and human mobility data. *Journal of Spatial Information Science*, 2020(20):73–93, 2020.

[32] Cheng J. Li, H. and X. Zhao. Deep learning in real estate: Predicting property prices with deep feature interactions. *Expert Systems with Applications*, 146:113150, 2020.

[33] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pages 4768–4777, 2017.

[34] Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad C, Ganesh Ramakrishnan, Micah Goldblum, and Colin White. When do neural nets outperform boosted trees on tabular data? *Advances in Neural Information Processing Systems*, 36, 2024.

[35] Charles Nelson Miller, Rol Rl, William Taylor, et al. Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417, 1970.

[36] Clara Petersen et al. Spatiotemporal extensions of cnps for dynamic data modeling. *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[37] Sherwin Rosen. Hedonic prices and implicit markets: product differentiation in pure competition. *Journal of Political Economy*, 82(1):34–55, 1974.

[38] Shriyank Somvanshi, Subasish Das, Syed Aaqib Javed, Gian Antariksa, and Ahmed Hossain. A survey on deep tabular learning. *arXiv preprint arXiv:2410.12034*, 2024.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.

[40] Lei Wang et al. Improving cnps with hierarchical latent variables. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2021.

[41] Zhang S. Wang, L. and Y. Li. Graph neural networks for real estate price prediction: capturing spatiotemporal dependencies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[42] Minho Yoo et al. Adaptive cnps for dynamic systems. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[43] H. Yu, Y. Wei, and L. Sun. Prediction of housing prices based on machine learning methods. In *Applied Mechanics and Materials*, volume 651, pages 3843–3847. Trans Tech Publications Ltd, 2014.

[44] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. Deep sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3391–3401, 2017.