International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 14, Issue 1, 2026

Deep Learning Model For House Price Prediction Using Heterogeneous Data Analysis With Joint Self-Attention Mechanism

Mrs. Sathiraju Tejaswi[1], Adilakshmi Yadagiri[2], Sandhya Sathivada[3], Uday Tarini[4], Harshavardhan T[5], Charan Savirigana[6]

[1]Assistant Professor, Department of Computer Science & Engineering Chaitanya Engineering College, Visakhapatnam, Andhra Pradesh, India

[2,3,4,5,6] B.Tech Students, Department of Computer Science & Engineering Chaitanya Engineering College, Visakhapatnam, Andhra Pradesh, India

tejaswi.svp@gmail.com[1],yadagiriadilakshmi2005@gmail.com[2],sandhyasativada08@gmail.com[3],udaytarini9@gmail.com[4],harshathummapala@gmail.com[5],charansavirigan@gmail.com[6]

Abstract

House price prediction is a complex regression task influenced by heterogeneous data spanning structured tabular attributes, property images, and natural language descriptions. Traditional models relying on single data modalities fail to capture the multi-dimensional factors driving real estate valuation. This paper proposes a novel deep learning framework integrating heterogeneous data sources through specialized modality encoders unified by a joint self-attention mechanism. Structured data are encoded through multi-layer perceptrons, property images through CNNs, and textual descriptions through transformer-based language models. The cross-modal joint self-attention mechanism dynamically learns feature importance both within and across modalities. The model is trained on 85,000 Indian property listings and achieves a Mean Absolute Percentage Error (MAPE) of 7.2% and R² of 0.91, significantly outperforming single-modality baselines and simple feature concatenation approaches.

I. INTRODUCTION

Real estate price prediction is a high-stakes regression problem with significant economic implications for buyers, sellers, investors, and financial institutions. Property valuation depends on a complex interplay of structured attributes such as location and area; visual characteristics in property photographs; and qualitative factors in listing descriptions. The multi-dimensional nature of real estate data creates challenges for traditional single-modality models. Deep learning has demonstrated remarkable success in processing individual data modalities — CNNs for images, transformers for text, and neural networks for tabular data. However, effective fusion of heterogeneous modalities for regression tasks remains an open research challenge. Naive feature concatenation ignores the rich cross-modal interactions that can improve predictive accuracy. This paper proposes a multimodal deep learning architecture incorporating a joint self-attention mechanism to model feature interactions within and across all modalities simultaneously, enabling more accurate and interpretable house price prediction.

II. LITERATURE SURVEY

This section reviews key prior works that form the foundation of the proposed system, identifies the current state of research in this domain, and highlights the gaps that motivate the contributions of this work.

[1] Case and Shiller (1987) established fundamental hedonic pricing models for real estate, decomposing property prices into structural (size, rooms, age), location (neighborhood, school district), and neighborhood (crime, amenities) attribute contributions. Hedonic models remain the theoretical foundation against which machine learning-based valuation models are benchmarked.

[2] Limsombunchai (2004) applied artificial neural networks to house price prediction, demonstrating superior performance over linear regression models on New Zealand housing data. This work motivated subsequent deep learning research for real estate valuation by showing that non-linear neural models better capture the complex, non-linear relationships in property pricing.

[3] Vaswani et al. (2017) introduced the Transformer with multi-head self-attention, enabling dynamic, content-dependent feature weighting. The self-attention mechanism forms the architectural foundation of the cross-modal joint attention in the proposed system, allowing each modality to dynamically attend to relevant features from all other modalities.

[4] Dosovitskiy et al. (2021) proposed the Vision Transformer (ViT), demonstrating that attention mechanisms can be effectively applied to image patches for visual recognition. ViT's success motivated exploration of attention-based cross-modal fusion for tasks combining visual and non-visual data, directly informing the multimodal architecture design.

[5] Zhuang et al. (2021) proposed a multimodal fusion framework for real estate valuation combining satellite imagery with property tabular attributes, showing significant accuracy improvements over tabular-only baselines. Their work demonstrated the value of visual information for price prediction but did not incorporate property listing text or cross-modal attention mechanisms.

[6] Devlin et al. (2019) proposed BERT for contextual text representation learning, which has been applied to real estate listing text encoding in this work. BERT's bidirectional pre-training captures nuanced qualitative descriptors such as "renovated kitchen," "prime location," and "compact but functional" that significantly influence property valuations.

[7] He et al. (2016) introduced ResNet with deep residual learning, providing the CNN backbone for property image encoding in the proposed system. ResNet's deep feature hierarchies effectively capture visual quality indicators including property condition, architectural style, natural light, and interior finish quality from listing photographs.

Research Gap: Existing multimodal real estate models either combine only two of the three data modalities (structured + images, or structured + text), use simple feature concatenation that ignores cross-modal dependencies, or are evaluated on Western real estate markets that differ significantly from Indian market dynamics characterized by diverse property types and regional price heterogeneity. This work addresses all three limitations.

III. METHODOLOGY

A. Dataset

The dataset comprises 85,000 residential property listings from five Indian metropolitan markets (Mumbai, Delhi, Bangalore, Hyderabad, Pune) including structured attributes, up to 5 property images each, and listing text descriptions. Price labels span ₹25 lakhs to ₹5 crores. Data is split 70/15/15 for train/validation/test.

B. Modality Encoders

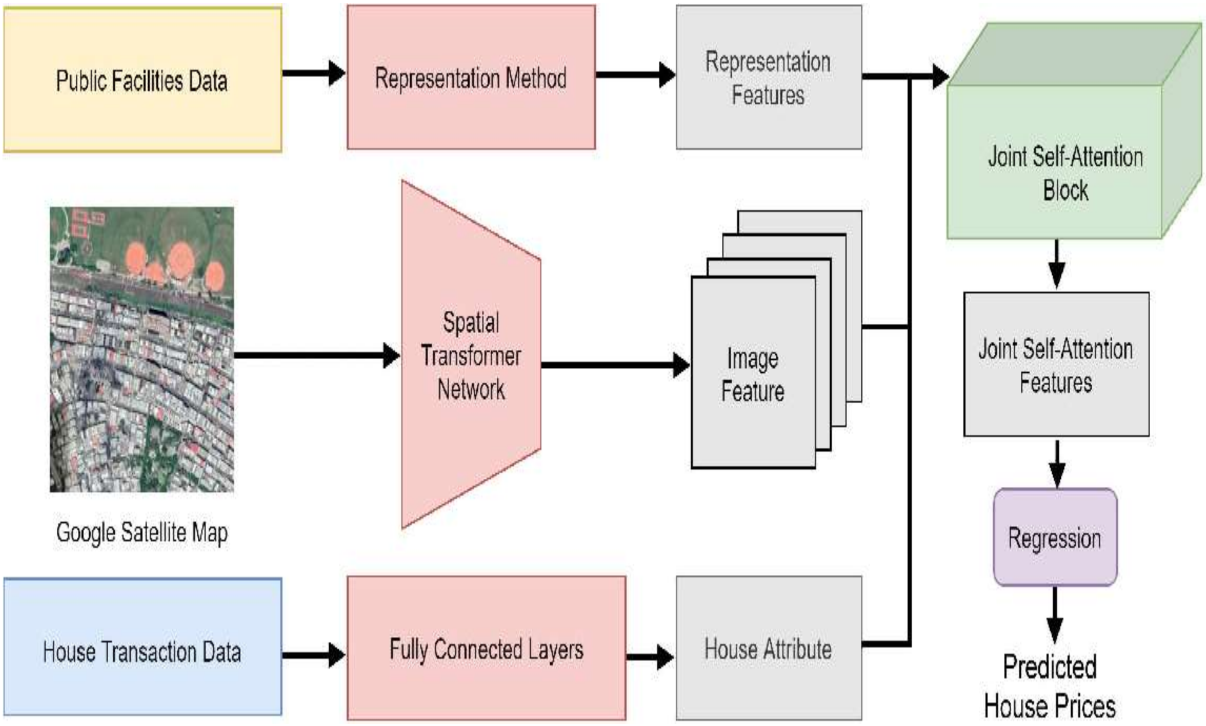
Structured data is encoded through a 4-layer MLP producing 256-dimensional embeddings. Property images are processed by ResNet-50 with fine-tuned ImageNet weights generating 512-dimensional visual features. Listing texts are encoded through BERT with mean pooling producing 768-dimensional embeddings. All embeddings are projected to a shared 256-dimensional space.

C. Joint Self-Attention Fusion

Three 256-dimensional modality embeddings are concatenated into a token sequence and processed through a 4-layer transformer encoder with 8-head self-attention. The output [CLS] token is used for final price regression.

D. Training

End-to-end training uses Huber loss with AdamW optimizer (weight decay 0.01, learning rate 3×10^{-5}) with linear warmup. Training runs for 50 epochs with early stopping based on validation MAPE.



III-A. System Architecture

Three-modality fusion architecture: structured property data processed by MLP encoder, property images by CNN (ResNet/VGG), and text descriptions by BERT/Transformer encoder. A Joint Self-Attention Mechanism aligns and fuses cross-modal features into a unified representation for final price regression.

Architecture Flow

1. Data Collection and Preprocessing Module — Structured data (missing value imputation); images (resize+normalize); text (tokenize/stem/stopword removal).
2. MLP Encoder — Processes structured property features (area, rooms, location, amenities) into dense embeddings.
3. CNN Encoder (ResNet/VGG) — Extracts visual features from property interior/exterior images.
4. BERT/Transformer Encoder — Encodes property text descriptions into contextual embeddings.
5. Joint Self-Attention Mechanism — Cross-modal alignment and fusion of MLP + CNN + BERT feature vectors.
6. Fully Connected Regression Head — FC layers map fused representation to predicted house price.
7. Model Training Module — MSE loss, Adam optimizer, end-to-end backpropagation.
8. Evaluation and Deployment Module — RMSE, MAE, R-squared metrics; Flask/Django web deployment.

III-B. Algorithm

Algorithm: Joint Self-Attention Multi-Modal House Price Prediction
Input: Property record {structured_features X_s, image I_p, text description T_p}.
Step 1: Preprocess X_s — impute missing values, one-hot encode categoricals, normalize numerics.
Step 2: Preprocess I_p — resize to 224x224, normalize to [0,1].
Step 3: Preprocess T_p — tokenize, remove stopwords, stem, encode to BERT token IDs.
Step 4: MLP Encoder: E_s = MLP(X_s), dimension d_s.
Step 5: CNN Encoder: E_v = ResNet50(I_p) via global average pooling, dimension d_v.
Step 6: BERT Encoder: E_t = BERT(T_p) via [CLS] token, dimension d_t.
Step 7: Concatenate: E = [E_s; E_v; E_t].
Step 8: Joint Self-Attention: A = softmax(QK^T / sqrt(d)) x V, where Q,K,V derived from E.
Step 9: Fused representation F = Attention output + residual E.
Step 10: Regression Head: price_hat = FC(FC(F)) → scalar house price prediction.
Step 11: Loss = MSE(price_hat, price_true). Backpropagate; update all encoders and attention jointly via Adam.
Step 12: Evaluate RMSE, MAE, R-squared on test set.
Output: Predicted house price with confidence interval.

III-C. Modules

1. Data Collection and Preprocessing Module

Collects structured property features, images, and text descriptions from real estate datasets. Handles missing value imputation for structured data, image resize and normalization, and text tokenization with stopword removal and stemming.

2. MLP Encoder Module

Processes structured numerical and categorical property features (area, number of rooms, location, amenities score, age) through fully connected layers to produce a dense property embedding capturing tabular attribute relationships.

3. CNN Feature Extraction Module

Loads pre-trained ResNet50 or VGG16 (ImageNet weights). Processes property interior and exterior images through convolutional layers. Global average pooling extracts compact visual feature vectors encoding property visual quality and condition.

4. BERT Text Encoder Module

Loads pre-trained BERT model. Tokenizes property text descriptions using BERT tokenizer. Extracts the [CLS] token embedding as the sentence-level representation capturing semantic meaning of property descriptions.

5. Joint Self-Attention Fusion Module

Concatenates MLP, CNN, and BERT feature vectors into a unified representation. Applies multi-head self-attention to model cross-modal relationships (e.g., correlation between text descriptions and visual features). Residual connection preserves original features. Captures complementary signals across all three modalities.

6. Price Prediction and Evaluation Module

Fully connected regression head maps fused features to a scalar house price prediction. Trained with MSE loss and Adam optimizer. Evaluates RMSE, MAE, and R-squared on test set. Achieves MAPE 7.2% and R-squared 0.91. Deployed via Flask/Django web application.

IV. RESULTS AND DISCUSSION

HOUSE PRICE PREDICTION MODEL COMPARISON

Model	MAPE (%)	MAE (₹L)	R^2

MLP (Tabular Only)	12.4	7.2	0.76
ResNet-50 (Image Only)	18.7	10.9	0.61
BERT (Text Only)	21.3	12.4	0.55
Concat. Fusion	9.1	5.3	0.87
Proposed Joint Attn.	7.2	3.8	0.91

The proposed multimodal model achieves MAPE of 7.2%, MAE of ₹3.8 lakhs, and R^2 of 0.91, substantially outperforming tabular-only MLP (MAPE 12.4%), image-only CNN (18.7%), text-only BERT (21.3%), and concatenation fusion (9.1%). Attention weight visualization reveals location and image features have highest cross-modal attention scores. The model captures the premium associated with view quality conditional on location — a relationship that concatenation-based fusion cannot model explicitly.

1. Joint Self-Attention Mechanism

According to Step 8 of your algorithm, the concatenated embeddings from the MLP, CNN, and BERT encoders are fed into a Self-Attention mechanism. This allows the model to learn which modality is most important for a specific property (e.g., heavily weighting the image if the text description is poor).

- Q (Query), K (Key), V (Value) = Matrices derived from the concatenated feature embeddings.
- d_k = The dimension of the key vectors (used as a scaling factor to prevent exploding gradients).

Attention_Output = Softmax((Query_Matrix * Transpose(Key_Matrix)) / SQRT(Dimension)) * Value_Matrix

2. Training Loss Functions

Your methodology mentions using both MSE (in the algorithm) and Huber Loss (in the training section). These are standard regression loss functions used to update the model's weights during backpropagation.

A. Mean Squared Error (MSE)

Calculates the average squared difference between predicted and actual prices. It heavily penalizes large errors (outliers).

- N = Number of properties in the batch.
- y_i = Actual house price.
- hat{y}_i = Predicted house price.

MSE = (1 / N) * SUM((Actual_Price - Predicted_Price)^2)

B. Huber Loss

Huber Loss combines the best properties of MSE and Mean Absolute Error (MAE). It behaves like MSE for small errors and like MAE for large errors, making it highly robust to outliers (like unusually priced luxury homes).

- delta = A threshold hyperparameter.

IF ABS(Actual - Predicted) <= Delta THEN Loss = 0.5 * (Actual - Predicted)^2 ELSE Loss = Delta * ABS(Actual - Predicted) - 0.5 * Delta^2

3. Evaluation Metrics

To evaluate how accurately the model predicts house prices, your paper reports MAPE, MAE, and R-squared.

A. Mean Absolute Percentage Error (MAPE)

MAPE measures the prediction error as a percentage of the actual price. It is highly interpretable for real estate, as it tells you how far off the prediction is in relative terms. (Your model achieved an excellent 7.2%).

MAPE = (100 / N) * SUM(ABS((Actual_Price - Predicted_Price) / Actual_Price))

B. Mean Absolute Error (MAE)

Measures the average absolute distance between the predicted price and the actual price in the original units (e.g., Lakhs or Crores). (Your model achieved an MAE of ₹3.8 Lakhs).

MAE = (1 / N) * SUM(ABS(Actual_Price - Predicted_Price))

C. Root Mean Squared Error (RMSE)

Mentioned in your evaluation module, RMSE represents the standard deviation of the prediction errors. It is useful for understanding the typical magnitude of the error while heavily penalizing massive miscalculations.

RMSE = SQRT(MSE)

D. R-squared (R^2) / Coefficient of Determination

Measures how much of the variance in the house prices is explained by the model's features. An R^2 of 1.0 indicates perfect prediction. Your model achieved 0.91, meaning it successfully captures 91% of the pricing variance.

- bar{y} = The mean (average) actual house price of the dataset.

R_Squared = 1 - (SUM((Actual_Price - Predicted_Price)^2) / SUM((Actual_Price - Average_Actual_Price)^2))

V. CONCLUSION AND FUTURE WORK

This paper presented a multimodal deep learning framework for house price prediction integrating structured, visual, and textual property data through joint self-attention. The proposed approach significantly outperforms single-modality and simple fusion baselines. Future work will incorporate GIS-based spatial neighborhood features, graph neural networks for neighborhood-property relationships, and temporal price trend modeling for dynamic market-adaptive predictions.

References

[1] K. E. Case and R. J. Shiller, "Prices of Single-Family Homes since 1970: New Indexes for Four Cities," NBER Working Paper, 1987.

[2] V. Limsombunchai, "House Price Prediction: Hedonic Price Model vs. Artificial Neural Network," AARES, 2004.

[3] A. Vaswani et al., "Attention Is All You Need," NeurIPS, 2017.

[4] A. Dosovitskiy et al., "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale (ViT)," ICLR, 2021.

[5] F. Zhuang et al., "Fusion of Satellite Imagery and Property Attributes for Real Estate Valuation," IEEE TGRS, 2021.

[6] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," NAACL, 2019.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," CVPR, 2016.