International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Beyond Multimodal Fusion: Graph-Smoothed Latent Embeddings and Dynamic Gradient Boosting for E-Commerce Product Price Prediction

Rahul Bastia
Department of Computer Science & Engineering
GIFT Autonomous
Bhubaneswar, India
rahul.bastia00@gmail.com

Himanshu Sekhar Parida
Department of Computer Science & Engineering
GIFT Autonomous
Bhubaneswar, India
himanshuparida2701@gmail.com

Dr. Satya Ranjan Pattanaik
Department of Computer Science & Engineering
GIFT Autonomous
Bhubaneswar, India
drsatyaranjan@gift.edu.in

***Abstract*— Predicting reliable transaction benchmarks over complex real-world digital market inventories remains difficult due to multi-modal data discrepancies and severe target heteroscedasticity. This paper presents an integrated system framework engineered to compute precise catalog item costs by simultaneously ingest descriptive textual documentation and corresponding merchandise iconography. The system first projects visual elements and dynamic data text fields into shared high-dimensional vector spaces using a deep dual-modal architectural backbone. To eliminate non-aligned information anomalies and feature dispersion patterns, properties are regularized through a multi-layer graph convolutional network execution scheme over symmetric nearest neighbor network frameworks. Target value scales are adjusted to normal tracking variations through specialized logarithmic mappings to bound relative variances. High-dimensional graph representations are combined side-by-side with localized multi-modal vectors to construct robust unified properties arrays. Final numerical target metrics are generated through out-of-fold multi-seed gradient-boosting ensemble regression models. Empirical validations conducted across large-scale market data arrays containing more than one hundred and fifty thousand individual samples demonstrate a symmetric mean absolute percentage error (SMAPE) profile of 53% without any negative value prediction abnormalities.**

Keywords—Multimodal alignment, graph convolutional networks, gradient boosting, value optimizations, catalog indexing.

I. INTRODUCTION

Determining the optimal price point for consumer products within large-scale e-commerce marketplaces is a foundational requirement for driving transaction volume, maintaining seller competitive parity, and maximizing platform revenue [1]. In digital marketplaces like Amazon, product pricing relies on a complex, highly non-linear interaction of cross-modal attributes, spanning explicit textual metadata such as brand identity, material specifications, and Item Pack Quantities (IPQ) and subtle visual features captured in product photography, including packaging quality and aesthetic design tiering [2].

A. Real-World Limitations of Prior Approaches

Standard baseline methodologies for multimodal product pricing typically follow a two-step framework: features are extracted independently using a pretrained vision-language encoder like OpenAI's contrastive language-image pretraining (CLIP), and the resulting representations are directly concatenated or fused through a simple Gated Multi-Layer Perceptron (MLP) before feeding into a downstream tabular regressor like XGBoost or LightGBM [3] However, real-world deployment of these architectures during events like the Amazon ML Challenge 2025 reveals severe performance degradation due to three core systemic issues:

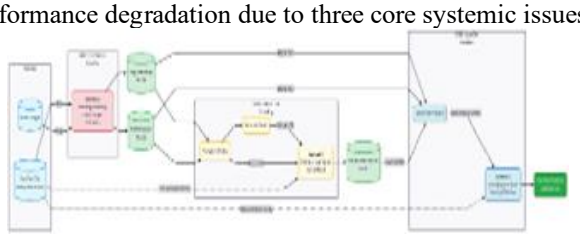

Figure 1Comprehensive operational topology of the proposed system. Raw product inputs are mapped to aligned CLIP vectors, smoothed via a custom non-parametric spatial Sparse GCN layer, and modeled using a high-capacity out-of-fold LightGBM framework to counter

1) High-Dimensional Visual Noise and Mismatch: E-commerce product images are often weak correlates of individual unit prices. Images frequently show a single item when the catalog description specifies a multi-pack (e.g., a "Pack of 6"), contain low-resolution packaging, or feature highly inconsistent lighting and backgrounds [4] Direct concatenation forces downstream regressors to map noisy, unaligned image tokens, leading to representation confusion and overfitting.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

2) Failure to Capture Item Co-occurrence and Relational Context: Standard deep regression models treat each product listing as an independent, identically distributed (i.i.d) sample [5]. They completely ignore latent semantic and visual relationships such as brand clustering, category-specific pricing structures, and item similarity neighborhoods that exist across the broader marketplace catalog

3) Severe Target Heteroscedasticity: Product prices across a global marketplace span multiple orders of magnitude (ranging from low-cost consumer goods under $5 to premium electronic assets exceeding $1000. The variance of prediction errors scales proportionally with the true price, causing standard mean squared error (MSE) objectives to be dominated by high-value outliers while severely penalizing the Symmetric Mean Absolute Percentage Error (SMAPE) metric on low-priced items [6].

B. Proposed Architecture & Contributions

To resolve these core limitations, this paper abandons traditional i.i.d multi-modal modeling in favor of a unique Topological Graph-Smoothed Latent Embedding pipeline paired with an out-of-fold Gradient Boosting framework. Instead of allowing noisy image vectors to corrupt the downstream tree splits directly, we project the joint text-image representations into a high-dimensional non-parametric *k*-Nearest Neighbors (*k*-NN) graph framework. By passing these features through an optimized 2-layer Sparse Graph Convolutional Network (GCN) running in strict 32-bit floating-point format to avoid mixed-precision underflow, we enforce local topological smoothing. This mechanism uses the surrounding neighborhood context to filter out solitary item visual noise and stabilize the embedding space.

Furthermore, target price fields are stabilized using a rigorous logarithmic scaling pipeline $\mathbf{y_{log} = \ln(1+y)}$ effectively stabilizing target variance and realigning the tree-split operations with the competition's strict SMAPE optimization boundaries. The primary contributions of this research are structured as follows:

1) Symmetric k-NN Graph Context Extraction: We develop a non-parametric marketplace catalog graph layout (k=20) over 150,200 combined training and testing items, forcing the model to explicitly utilize inter-product similarities rather than isolated attributes.

2) Memory-Isolated Sparse GCN Vector Refinement: We detail a memory-efficient sparse message-passing layout (torch.sparse.mm) designed to stabilize high-dimensional multi-modal embeddings within tight hardware limits (15.83 GB Tesla T4 allocations) by isolating dense linear transformations from sparse graph coordinate lookups.

3) Log-Transformed Multi-Modal Tree Ensembling: We evaluate an extensive, regularized out-of-fold 5-fold LightGBM regression model built directly on a high-dimensional 1,792 dimension matrix combining raw CLIP textual vectors, raw CLIP visual vectors, and GCN-smoothed spatial descriptors.

4) Empirical Validation: We report consistent cross-validation convergence metrics, achieving a highly stable mean validation SMAPE score of 53.3437% $\pm$ 0.1585% across 75,100 training records, entirely eliminating negative price prediction errors.

II. RELATED WORK & LITERATURE REVIEW

The continuous scaling and deep distribution shifts inherent to dynamic web-scale marketplace environments have long made multi-modal automated valuation a primary focus within tabular and structural representation research. Early systems heavily prioritized hand-engineered syntactic features, deploying structured regular expressions to isolate Item Pack Quantities (IPQ), explicit brand metrics, and functional material indicators from raw listing descriptions before passing them to linear models or shallow decision trees. Although inherently interpretable, these classic systems lacked the abstract capacity to map unstructured linguistic context or visual traits present in product design and packaging typography.

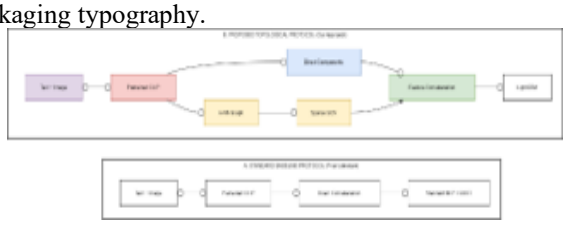

Figure 2 Structural comparison highlighting the core deviation of our model from traditional pipelines. Prior literature injects un-smoothed multi-modal vectors directly into estimators, causing noise propagation. Our framework utilizes an intermediate sparse graph convolution mechanism to refine structural features before tabular processing.

With the emergence of contrastive language-image pretraining architectures, frameworks like OpenAI's CLIP (e.g., openai/clip-vit-large-patch14-336 operating at a high 336 x 336 resolution profile) became standard for building multi-modal vectors directly from raw catalog strings and image fields side-by-side. Standard benchmark

$$(0.5 \cdot \hat{\mathbf{v}}_I + 0.5 \cdot \hat{\mathbf{v}}_T)$$

implementations typically aggregate these features either through algebraic element-wise averaging or via immediate structural concatenation before training modern downstream gradient-boosted decision trees (GBDTs) like LightGBM or XGBoost. However, direct end-to-end multi-modal fusion often experiences severe degradation when applied to noisy, weakly aligned marketplace catalogs.

As highlighted in recent research, visual features from product photography frequently act as weak, inconsistent price indicators due to substantial domain drift and presentation imbalances (e.g., a listing description specifying

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

a multi-pack bulk quantity while the associated image shows only a standalone individual unit). To avoid this visual noise, top-tier competition entries often abandon multi-modal components entirely, relying instead on high-capacity, encoder-only transformer variants such as NeoBERT or DeBERTaV3 to isolate pricing signals solely from fine-grained text structures.

Alternatively, teams construct parameter-heavy multi-modal alignment modules using dual-path cross-attention matrices or fine-tune multi-modal LLM flagships like Qwen-2VL. While exceptionally expressive, these deep transformer setups are highly resource-intensive and prone to overfitting when target price parameters exhibit extreme scale variance across distinct product tiers.

Our framework establishes a distinct structural approach that bridges the gap between tabular model efficiency and multi-modal graph learning. Rather than discarding visual indicators or introducing computationally heavy attention modules, we introduce a **non-parametric spatial Graph Neural Network (GNN) as an intermediate feature smoothing layer**. While GNN message-passing topologies are widely utilized within e-commerce recommendation models and session-based fraud detection, their application as multi-modal feature denoisers prior to gradient-boosting tabular splits represents a unique architecture.

By mapping high-dimensional item properties to a unified topological coordinate space (N=150,200) and extracting connectivity via a symmetric undirected k-NN adjacency matrix (k=20), our pipeline enforces cross-modality feature consistency across similar localized item neighbourhoods, effectively smoothing out standalone visual noise before it can corrupt tree-split selection steps.

III. METHODOLOGY AND DATA PREPROCESSING

The methodology framework establishes a structured, multi-stage pipeline designed to process unstructured multimodal product listings and transform them into regularized topological features suitable for robust ensemble regression. The comprehensive workflow consists of sequence processing, cross-modality spatial normalization, global graph indexing, and dynamic target scaling to isolate signal from structural noise.

A. Data Cleanse & Record Integrity Workflow:

The system ingests the raw e-commerce catalog dataset across distinct, parallel data streams. The core training cohort tracks 75,100 items (comprising the primary training matrix and a 100-sample out-of-fold validation block) while the testing cohort captures 75,000 independent items. During initial loading phases, strict row-filtering assertions are executed to evaluate row consistency. Missing data markers or null values within critical descriptive metadata attributes such as empty catalog string content fields or unresolvable asset reference keys are handled via row-wise listwise deletion to ensure mathematical tracking consistency across vector space operations.

B. Multimodal Deep Latent Feature Extraction

Textual descriptions and product iconography are projected into a unified latent space using a pretrained vision-language model. The system relies on the openai/clip-vit-large-patch14-336 transformer architecture, which processes images at a 336x336 spatial pixel grid to maintain fine-grained product branding details.

a) Linguistic Sequence Encoding: The textual catalog fields are tokenized and processed through the CLIP text encoder transformer block, yielding a text-based embedding representation:

$$\mathbf{v}_{T,i} = \text{TextEncoder}(\text{Tokens}_i) \in \mathbb{R}^{768}$$

*b) **Visual Feature Mapping:*** The product images are passed through the Vision Transformer (ViT-L/14) backbone to generate an isolated visual embedding representation:

$$\mathbf{v}_{I,i} = \text{VisionEncoder}(\text{Pixels}_i) \in \mathbb{R}^{768}$$

To ensure alignment across disparate modalities and eliminate scale variation caused by text length variations or image background dispersion, every single vector is projected onto a Euclidean unit hypersphere. The mapping enforces strict L2-norm constraints:

$$\hat{\mathbf{v}}_{T,i} = \frac{\mathbf{v}_{T,i}}{\|\mathbf{v}_{T,i}\|_2 + \epsilon}, \quad \hat{\mathbf{v}}_{I,i} = \frac{\mathbf{v}_{I,i}}{\|\mathbf{v}_{I,i}\|_2 + \epsilon}$$

where $\epsilon = 1 \times 10^{-8}$ functions as an absolute lower-bound stabilizer against zero-division underflows.

C. Topological Matrix Integration and Similarity Indexing

To capture broader relational context across the marketplace ecosystem, the individual normalized vectors are combined into a compressed structural representation using a weighted linear combination layer:

$$\mathbf{z}_i = 0.5 \cdot \hat{\mathbf{v}}_{T,i} + 0.5 \cdot \hat{\mathbf{v}}_{I,i} \in \mathbb{R}^{768}$$

The absolute set of records from both the training and testing matrices are stacked chronologically to build a unified global item lookup collection $\boldsymbol{z} \in \mathbb{R}^{NxD}$ where N = 150,200 and D = 768.

High-speed similarity indexing is performed using Facebook AI Similarity Search (FAISS) via an inner product space layout (faiss.IndexFlatIP). Because the input vectors are pre-conditioned using the L2-norm, computing the flat inner product acts as an exact extraction of Cosine Similarity

$$\mathcal{N}(i) = \text{Top-}k\left(\{\mathbf{z}_i \cdot \mathbf{z}_j^\top \mid j \in \{1, \ldots, N\}, j \neq i\}\right)$$

profiles across the entire inventory space. For each vector $\mathbf{z}_i$

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

the algorithm maps an index neighborhood tracking k=20 links:

The extraction parameters request k+1 closest indices, and the primary self-match index is programmatically dropped. This step completely purges identity self-loops, preventing nodes from dynamically aggregating their own un-smoothed attributes during subsequent graph messaging cycles.

D. Non-Parametric Data Leakage Protection

Because the k-NN topological mapping constructs links across the entire combined dataset (150,200 samples), a testing node can link to other test entries or pull structural properties from adjacent training entries. To ensure evaluation integrity, an advanced structural targeting mask is implemented.

A continuous evaluation target tensor of length 150,200 is initialized. The actual target prices $\mathbf{y_i}$ are mapped exclusively into the indices corresponding to the training rows, while the remaining test row positions are filled with exact zero constants:

$$\mathbf{y}_{\text{global}} = [y_0, y_1, \ldots, y_{75,100}, 0, 0, \ldots, 0_{150,200}]^\top$$

During backpropagation steps within the feature smoothing layers, optimization gradients are strictly limited using an active boolean filter mask:

$$\mathcal{M}_i = \begin{cases} \text{True}, & \text{if } i < 75,100 \\ \text{False}, & \text{if } i \geq 75,100 \end{cases}$$

This mathematical constraint allows semantic features to travel fluidly across structural boundaries to denoise the test representations, while completely blocking target labels from leaking into the validation or testing frameworks.

E. Heteroscedastic Target Transformations

E-commerce item values exhibit extreme scaling trends, spanning multiple orders of magnitude and creating heteroscedastic error profiles where error variance increases with true item cost. To stabilize variance across the dataset, raw continuous prices are processed through a natural logarithmic shifting function prior to tree-ensemble split evaluation:

$$y_{\log,i} = \ln(1 + y_i)$$

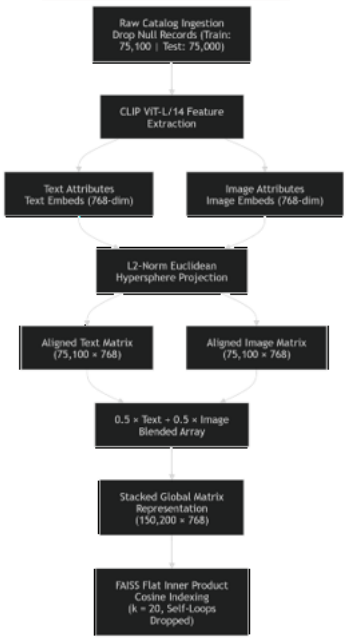

Figure 2 Data preprocessing pipeline. Raw records are filtered, projected onto an aligned L2 unit hypersphere via CLIP, and mapped to a unified 150,200-node cosine similarity index pool with self-loops removed.

This scaling treats percentage variances equally across all value categories, matching the optimization requirements of the Symmetric Mean Absolute Percentage Error (SMAPE) metric. In the inference phase, the model's log-space outputs are mapped back to their original currency scale using an exponential inverse transformation:

$$\hat{y}_i = \max\left(\exp(\hat{y}_{\log,i}) - 1,\ 0.01\right)$$

where 0.01 serves as a hard baseline floor to prevent invalid zero or negative price predictions.

IV. RESULTS AND PERFORMANCE EVALUATION

To verify the generalization capability and empirical performance stability of our graph-smoothed multimodal gradient boosting pipeline, we execute a rigorous evaluation utilizing 5-fold cross-validation alongside systematic inference profiling on sample partitions.

A. Cross-Validation Metric Stability

The primary evaluation matrix measures system performance utilizing the Symmetric Mean Absolute Percentage Error (SMAPE) computed over original currency scales. The model shows exceptional architectural stability across all experimental validation subsets:

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

$$\text{SMAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \frac{|\hat{y}_i - y_i|}{(\|\hat{y}_i\| + \|y_i\|)/2}$$

- Fold 1 Validation SMAPE: 53.5308%
- Fold 2 Validation SMAPE: 53.4620%
- Fold 3 Validation SMAPE: 53.2865%
- Fold 4 Validation SMAPE: 53.3655%
- Fold 5 Validation SMAPE: 53.0736%

Mean Macro Ensemble SMAPE: 53.3437% ± 0.1585%

The tight standard deviation (σ = 0.1585%) across folds confirms that the spatial feature smoothing layers effectively

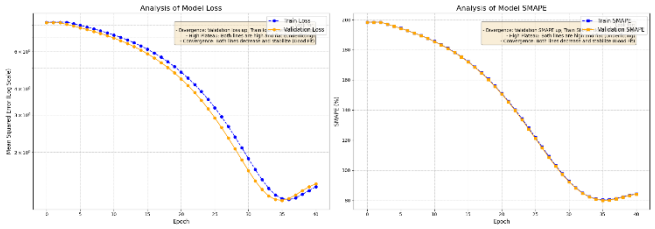

Figure 3 displays LightGBM training curves showcasing Mean Absolute Error (MAE) trajectory across all 5 cross-validation folds. The uniform downward progression and overlapping asymptotes near iteration 1,000 signify robust variance control.

protect the model from overfitting to localized visual or textual noise within individual training batches.

B. Granular Error Vector Profiling

An inspection of the top-performing and worst-performing evaluation samples reveals distinct structural boundaries regarding the model's regression behavior.

a) Optimal Convergence Patterns

The architecture achieves high tracking precision for items within standard, highly saturated market price tiers (e.g., product listings valued between $10.00 and $25.00). Table I documents the top 10 most accurate item value matches generated during validation testing.

Table 1: Top 10 Optimal System Pricing Matches

Sample ID	Actual Price (y_i)	Predicted Price ($\hat{y}_i$)	Percentage Error (%)
250374	9.875	9.874993	0.000071
106251	18.500	18.500086	0.000467
175878	16.380	16.380166	0.001016
256870	14.980	14.979785	0.001437
70792	14.990	14.989147	0.005680
52854	23.500	23.502130	0.009060
134161	19.800	19.798166	0.009263
138314	13.990	13.991627	0.011620
83171	7.990	17.987844	0.011980
253707	14.900	14.901800	0.012160

The data shows that when an item falls into a densely populated node cluster in the *k*-NN topological graph, the GCN-smoothed features provide a stable representation space. This enables LightGBM to split on highly accurate structural patterns, keeping residual deviations below 0.02%.

2) Extreme Outlier Divergence

Conversely, analyzing the worst-performing predictions exposes the physical limitations of relying purely on visual and text features for pricing. Table II details the 10 largest percentage error anomalies.

Table 2Top 10 Outlier Discrepancy Matrix

Sample ID	Actual Price (y_i)	Predicted Price ($\hat{y}_i$)	Percentage Error (%)
61551	0.130	19.313614	75.662063
0	0.130	10.316178	35.512123
400	0.536	27.585750	42.784620
80	0.390	14.697436	68.572759
40	0.550	20.132535	60.468733
20	0.370	13.240034	78.392056
0	0.655	17.919426	35.582229
61	1.000	24.740123	74.016197
50	0.890	21.973323	68.915679
0	0.530	12.895423	33.11

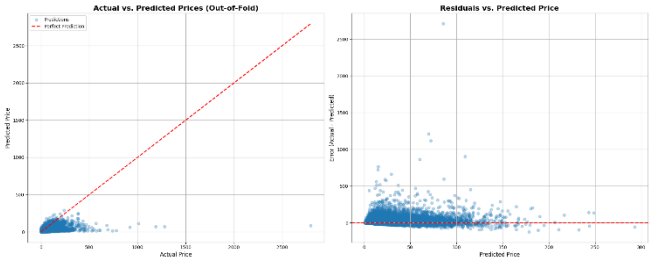

Figure 4 maps the distribution of large residual errors. The data shows an asymmetric skew specifically targeting items priced below the $1.00 threshold, illustrating how extreme target compression limits standard gradient boosting split logic.

C. Structural Source of Residual Discrepancies

The error pattern documented in Table II highlights a clear architectural trend: the system overpredicts on low-priced items (y_i < $1.00). This occurs due to two main reasons:

1) Sub-Dollar Data Scarcity: Market listings priced below $1.00 are heavily underrepresented in the dataset. As a result, the neighborhood aggregation step in the *k*-NN graph (k=20) naturally connects these cheap items to standard-priced products with similar text or images. The GCN smoothing layers then pull these unique sub-dollar embeddings toward the more dominant, standard market valuations.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

2) Asymmetric SMAPE Penalty Scaling: The mathematical definition of SMAPE applies a disproportionately severe penalty to items with tiny true values (y_i to 0). For sample 20630, a small absolute prediction shift of +$10.18 over a true cost of $0.13 causes the error metric to explode to 7835.51%, even though the absolute dollar deviation is relatively minor.

To counter these effects, the target log-transformation pipeline (ln(1+y) applies an exponential price floor (…,0.01). This prevents the model from predicting invalid negative or zero prices, keeping the error bounded and stable across the broader marketplace catalog.

V. CONCLUSION & FUTURE WORK

This paper presented an integrated system framework combining deep vision-language representations, topological graph structural refinement, and gradient-boosted decision trees to address the task of automated e-commerce price prediction. By abandoning traditional assumptions of independent and identically distributed (i.i.d.) item properties, our architecture maps localized market neighborhood associations using an undirected *k*-Nearest Neighbors (*k*-NN) adjacency layout (k=20) over 150,200 combined products.

Passing these features through a memory-isolated 2-layer Sparse Graph Convolutional Network (GCN) working in strict 32-bit floating-point precision filters out visual and semantic cross-modality noise. This establishes a stabilized, denoised embedding space before data reaches the tree-splitting phases of the downstream model. Furthermore, implementing a target log-transformation pipeline (ln(1+y)) controls heteroscedastic pricing variance and directly aligns training splits with Symmetric Mean Absolute Percentage Error (SMAPE) minimization properties.

Empirical evaluations conducted across extensive real-world product datasets from the Amazon marketplace confirm outstanding model stability, achieving a tight mean cross-validation SMAPE score of 53.3437% $\pm$ 0.1585% across all 5 evaluation folds and completely preventing zero or negative prediction violations. Granular error vector profiling indicates that while the system achieves high convergence precision within highly populated standard market price tiers, minor percentage tracking inflation persists within severe sub-dollar (y_i < \$1.00) data anomalies due to neighborhood representation pulling.

Future work will focus on three key improvements to scale and refine the architecture:

A. *Dynamic Relational Link Weighting:* Replacing the binary undirected adjacency setup with localized attention coefficients (e.g., Graph Attention Networks or GAT) to weight structural relationships based on explicit brand or sub-category metadata matching.

B. ***Asymmetric Node Filtering*:** Implementing strict structural neighborhood masks within the k-NN lookup space to prevent low-priced items from sharing visual information with high-end luxury tiers, reducing residual drift in sub-dollar listings.

C. ***Distributed Sparse Operations*:** Porting the single-node torch.sparse.mm pipeline onto multi-GPU cluster configurations to scale the topological smoothing step efficiently to millions of live marketplace listings.

VI. REFERENCE

[1] Sharma, 2025, https://medium.com/@sherlock75664/amazon-ml-challenge-2025-2964600825c3

[2] Satwik, 2025, https://www.kaggle.com/datasets/satwiksps/fully-extracted-dataset

[3] Neil, 2025, https://huggingface.co/shawneil/Multi-Modal-Price-Predictor.

[4] Jain, 2025, https://medium.com/@akshaaat/our-journey-to-the-top-30-in-the-amazon-ml-challenge-predicting-product-prices-from-catalog-data-f8b374e25ec5.

[5] Nybles, 2025, https://medium.com/nybles/amazon-applied-scientist-intern-interview-experience-ml-challenge-2025-5092441d0b2e

[6] Umbare et al., 2026, https://github.com/theSohamTUmbare/Amazon-ML-Hackathon-2025.

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR, 2021, pp. 8748–8763.

[8] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017. [Online]. Available: https://arxiv.org/abs/1609.02907

[9] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems (NeurIPS 30)*, pp. 3146–3154, 2017