Advanced Large Language Model Ensemble for Multimodal Customer Identification in Banking Marketing

Yanjun Dai *, Haoyang Feng, Zhuqi Wang, Yuan Gao

Posted Date: 12 June 2025

doi: 10.20944/preprints202506.0994.v1

Keywords: Large Language Model; Multimodal Network; Customer Identification; Ensemble Learning; Banking Marketing

 doi:10.20944/preprints202506.0994.v1

Article

Advanced Large Language Model Ensemble for Multimodal Customer Identification in Banking Marketing

Yanjun Dai [1,*], Haoyang Feng [2], Zhuqi Wang [3] and Yuan Gao [4]

[1] Brandeis University, Waltham , USA
[2] Duke University, Durham, USA; brianmaga2024@gmail.com
[3] Washington University in St. Louis, Missouri, USA; zhuqi.wang@outlook.com
[4] Boston University, Boston, USA, xyan56379@gmail.com
* Correspondence: yanjundai0000@gmail.com

Abstract: This study proposes the Advanced Large Language Model Ensemble Multimodal Network (ALIMN) to improve target customer identification in banking marketing. The framework combines Large Language Models (LLMs) with a multi-branch deep ensemble structure to analyze both structured and unstructured financial data. The model uses LLMs for semantic embedding, fine-tuning, and attention mechanisms, which improve customer identification. The results show that ALIMN performs better than traditional and deep learning models. Future work will focus on improving LLM adaptation, optimizing efficiency, and applying it to areas like financial risk control and personalized recommendations.

Keywords: large language model; multimodal network; customer identification; ensemble learning; banking marketing

1. Introduction

The banking industry has changed a lot in recent years because of digital technologies. Financial institutions now focus on identifying high-value customers to offer personalized services. Traditional methods, based on demographic or transactional data, do not fully capture customer behavior. With big data and machine learning, banks can now analyze both structured and unstructured data, giving better insights into customer preferences.

Talaat et al. [1] introduced a hybrid model integrating deep learning, explainable AI, and RFM analysis for customer segmentation, enhancing targeting and behavioral insights crucial for financial regulation.

Potluri et al. [2] utilized machine learning for personalized marketing in finance, demonstrating that improved segmentation enhances targeting, retention, and revenue.

Tian Jin [3] introduced attention-based temporal convolutional networks and reinforcement learning for supply chain delay prediction, which can also be used for customer segmentation in banking. This method improves decision-making, helping better resource allocation and customer relationship management.

The proposed ALIMN combines both structured and unstructured data. It uses Large Language Models (LLMs) to analyze complex financial data, like transaction histories and financial documents. By applying domain-specific fine-tuning and attention mechanisms, ALIMN improves the understanding of customer data and provides more accurate targeting. The multi-branch deep ensemble structure further increases prediction accuracy, offering a more comprehensive solution for customer segmentation in banking.

 doi:10.20944/preprints202506.0994.v1

2. Related Work

Recent studies show that machine learning is increasingly used in customer segmentation for banking. Pandey et al. [4] demonstrated that machine learning models, such as clustering and supervised learning, provide more precise insights than traditional methods, improving targeted marketing.Tang et al. [5] introduced Box Adjuster, a reinforcement learning-based method that enhances OCR accuracy by optimizing text bounding boxes.Feng et al. [6] proposed DocPedia, a large multimodal model leveraging frequency-domain processing for superior OCR-free document understanding.Tang et al. [7] proposed a transcription-only text spotting method using query-based learning and audio annotations to reduce annotation costs.

Lu et al. [8] present an automated image-based method for extracting pavement texture and predicting MTD with high accuracy ($R^2 = 0.9858$).Tang et al. [9] introduced a transformer-based scene text detection method using feature sampling and grouping to enhance efficiency and eliminate post-processing, achieving state-of-the-art performance.Liu et al. [10] introduced SPTS v2, a single-point annotation-based scene text spotting framework that enhances efficiency and achieves state-of-the-art performance.Zhao et al. [11] proposed TextHarmony, a multimodal generative model leveraging Slide-LoRA for unified visual text comprehension and generation.

Julian and Hariprasath [12] emphasized the role of clustering algorithms in optimizing customer segmentation and marketing outcomes. Dan et al. [13] develop a deep learning-based multiview stereo method for asphalt pavement texture evaluation, achieving stable accuracy (IoU = 0.77) as a lightweight alternative to traditional techniques. Tian Jin [14] applied ensemble models for sales forecasting, enhancing segmentation accuracy in banking.Tang et al. [15] introduced MTVQA, a multilingual TEC-VQA benchmark addressing visual-text misalignment and performance gaps in state-of-the-art models.

Lastly, Dan et al. [16] propose an image-driven system for predicting pavement aggregate gradation, integrating deep learning and interactive processing, achieving high accuracy $R^2 > 0.99$ for quality assessment.Tang et al. [17] introduced TextSquare and Square-10M, a large-scale dataset that enhances text-centric VQA, surpassing state-of-the-art models. Zhao et al. [18] proposed E2STR, a multi-modal in-context learning model for scene text recognition, enabling training-free adaptation with state-of-the-art performance.Tang et al. [19] organized the first character recognition competition for street view shop signs, detailing tasks, datasets, and winning solutions.

3. Methodology

We propose the LLM Integrated Multi-Modal Network (ALIMN). This framework combines large language model (LLM) features with a multi-branch deep ensemble structure. It captures complex patterns in different types of financial data. ALIMN uses LLMs for extracting semantic features. It includes a contextual fusion module and a multi-model ensemble. These parts process structured and unstructured data together. The design applies special transformation layers and equations. This improves prediction accuracy and generalization in banking campaign response tasks. The system pipeline is shown in Figure 1.

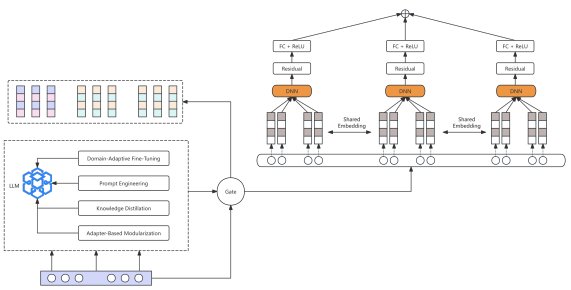

Figure 1. The LLM Integrated Multi-Modal Network.

 doi:10.20944/preprints202506.0994.v1

3.1. Extended LLM Integration

Beyond the basic transformer-based embedding, the following advanced strategies are introduced to further enrich the LLM Feature Extractor and adapt it to the financial domain:

3.1.1. Domain-Adaptive Fine-Tuning

The pre-trained LLM is powerful. However, financial and banking texts have unique terms and meanings. To handle this, we fine-tune the model using banking documents, such as product descriptions and financial news. This process updates Θ_{LLM} to learn financial terms and relationships. For each token t_i in a financial text $\mathcal{D}$, we minimize:

$$\mathcal{L}_{\text{DA}} = -\sum_{i=1}^{N} \log\, p\big(t_i \mid t_{<i}; \Theta_{\text{LLM}}\big), \tag{1}$$

where $t_{<i}$ includes all tokens before t_i, and N is the total number of tokens in the text.

3.1.2. Prompt Engineering and Instruction Tuning

When $\mathbf{x}$ has short text segments, such as customer notes, we use prompt engineering to help the LLM create more relevant embeddings. Instruction tuning makes the LLM respond better to financial prompts, like "Identify campaign-related interests." If $\mathbf{x}_{\text{prompt}}$ is the input text with a prompt r, we write:

$$\mathbf{e}_{\text{prompt}} = \text{LLM}\big([\mathbf{x}; r]; \Theta_{\text{LLM}}\big), \tag{2}$$

which gives a special embedding for later tasks.

3.1.3. Multi-Phase Training with Knowledge Distillation

We do not use a single fixed LLM checkpoint. Instead, we apply a multi-phase training method. First, a large teacher model, such as T5 or DeBERTa-v3, creates high-quality embeddings or intermediate outputs $z_i^{(\text{T})}$ for each input i. Then, a smaller student LLM generates $z_i^{(\text{S})}$ by copying these outputs. We minimize the Kullback-Leibler (KL) divergence:

$$\mathcal{L}_{\text{KD}} = \sum_{i=1}^{N} \text{KL}\big(p_{\text{teacher}}(z_i^{(\text{T})}),\, p_{\text{student}}(z_i^{(\text{S})})\big). \tag{3}$$

Here, p_{teacher} and p_{student} are the outputs of the teacher and student models. The pipeline of the teacher-student model is shown in Figure 2.

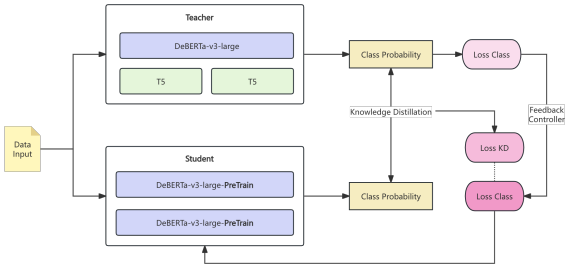

Figure 2. The Multi-Phase Training with Knowledge Distillation.

3.1.4. Adapter-Based Modularization

To process different data sources, such as regulatory text, marketing messages, and product descriptions, lightweight adapter modules are added to Transformer blocks. Let h_l be the output of the l-th Transformer layer. The adapter $\text{Adpt}_l(\cdot)$ is applied as:

$$h_l' = h_l + \text{Adpt}_l\big(h_l; \Theta_{\text{adpt}}\big), \tag{4}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 June 2025 doi:10.20944/preprints202506.0994.v1

where Θ_{adpt} represents the adapter-specific parameters. During inference, the model activates the right adapters based on domain tags linked to the input data. The pipeline of the adapter-based approach is shown in Figure 3.

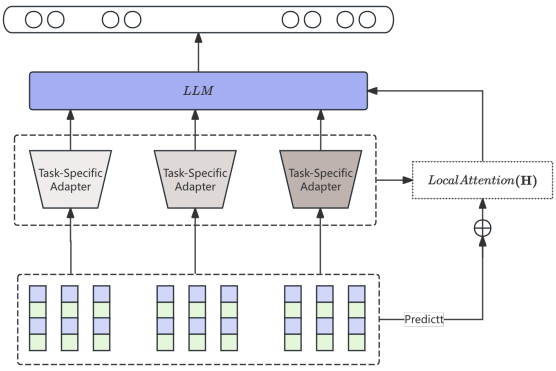

Figure 3. The Adapter-Based Modularization.

3.1.5. Hierarchical Attention for Hybrid Inputs

When $\mathbf{x}$ includes both text segments, such as tokenized sentences, and structured metadata, such as demographic codes, we use a hierarchical attention mechanism. Let $H_{\text{text}} \in \mathbb{R}^{T\times d}$ be the hidden states of text from LLM self-attention. Let $H_{\text{meta}} \in \mathbb{R}^{M\times d}$ be the embedding of M structured features. We compute:

$$\mathbf{A}_{\text{cross}} = \text{softmax}\Big(\frac{H_{\text{text}}\, H_{\text{meta}}^{T}}{\sqrt{d}}\Big), \quad H_{\text{fusion}} = \mathbf{A}_{\text{cross}}\, H_{\text{meta}}, \tag{5}$$

which captures cross-modal relationships. The fused representation H_{fusion} is then combined with H_{text} to create the final representation.

3.2. Contextual Fusion Module

This module combines semantic embeddings from the LLM with structured features $\mathbf{s} \in \mathbb{R}^{p}$. A fusion mechanism is used. A gating function learns the best combination:

$$\mathbf{z} = \sigma\Big(\mathbf{W}_f[\mathbf{e};\mathbf{s}] + \mathbf{b}_f\Big) \odot \mathbf{e} + \Big(1 - \sigma\Big(\mathbf{W}_f[\mathbf{e};\mathbf{s}] + \mathbf{b}_f\Big)\Big) \odot \mathbf{s}, \tag{6}$$

where $[\mathbf{e};\mathbf{s}]$ is the concatenation of $\mathbf{e}$ and $\mathbf{s}$. $\mathbf{W}_f \in \mathbb{R}^{q\times(k+p)}$, $\mathbf{b}_f \in \mathbb{R}^{q}$, $\sigma(\cdot)$ is the sigmoid activation function, and $\odot$ is element-wise multiplication. The gating function controls how features are mixed based on the input.

3.3. Multi-Branch Ensemble

The fused representation $\mathbf{z}$ is sent to a multi-branch ensemble. This structure captures different parts of the data using separate networks. There are M branches, each processing $\mathbf{z}$ with its own transformation:

$$\mathbf{h}_m = f_m(\mathbf{W}_m\mathbf{z} + \mathbf{b}_m), \quad m = 1, 2, \ldots, M, \tag{7}$$

where $f_m(\cdot)$ is the activation function for branch m, and $\mathbf{W}_m$, $\mathbf{b}_m$ are the weights and biases for that branch.

The outputs from all branches are combined using a weighted sum to get the final prediction:

$$\hat{y} = \sigma\left(\sum_{m=1}^{M} \alpha_m\, \mathbf{w}_m^{T}\mathbf{h}_m + b_{\text{out}}\right), \tag{8}$$

 doi:10.20944/preprints202506.0994.v1

where $\alpha_m \in [0,1]$ is the weight for branch m, with $\sum_{m=1}^{M} \alpha_m = 1$. $\mathbf{w}_m$ is a projection vector for branch m, and b_{out} is the output bias. The activation function $\sigma(\cdot)$ (such as sigmoid) scales the output for binary classification.

3.4. Regularization and Residual Connections

Residual connections are added to each branch to improve training stability:

$$\mathbf{h}_m = f_m(\mathbf{W}_m \mathbf{z} + \mathbf{b}_m + \mathbf{z}), \tag{9}$$

and an L_2 regularization term is applied to all learnable parameters Θ:

$$\mathcal{R}(\Theta) = \lambda \sum_i \|\Theta_i\|_2^2. \tag{10}$$

3.5. Loss Function

The training objective of ALIMN is to minimize a composite loss function that accounts for classification error, regularization, and auxiliary penalties to ensure robust learning. The overall loss is formulated as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{aux}}, \tag{11}$$

where each term is detailed as follows.

3.5.1. Binary Cross-Entropy Loss

For binary classification, the primary loss function is the binary cross-entropy (BCE) loss, defined over a dataset of N samples:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N} \sum_{i=1}^{N} \Big[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \Big], \tag{12}$$

where $y_i \in \{0,1\}$ represents the true label and $\hat{y}_i$ is the predicted probability obtained from the ensemble output in (8).

3.5.2. Regularization Loss

To prevent overfitting and promote generalization, an L_2 regularization term is added:

$$\mathcal{L}_{\text{reg}} = \lambda \sum_j \|\Theta_j\|_2^2, \tag{13}$$

where Θ_j denotes the j-th parameter group in the network and λ is the regularization coefficient.

3.5.3. Auxiliary Loss

An auxiliary loss is added to improve feature learning in the multi-branch ensemble. It aligns intermediate representations with the final prediction. If $\mathbf{h}_m$ is the output from the m-th branch, we define:

$$\mathcal{L}_{\text{aux}} = \frac{1}{M} \sum_{m=1}^{M} \|\mathbf{h}_m - \mathbf{z}\|_2^2, \tag{14}$$

which helps each branch stay consistent with the fused representation $\mathbf{z}$ from the Contextual Fusion Module.

3.6. Data Preprocessing

Optimized data preprocessing enhances ALIMN's performance, involving key steps:

 doi:10.20944/preprints202506.0994.v1

3.6.1. Normalization

Continuous features are standardized to zero mean and unit variance for consistency across different scales.

3.6.2. Categorical Encoding

Categorical variables are transformed using one-hot encoding to facilitate model compatibility.

3.6.3. Feature Augmentation

LLM-based feature extraction enriches input features, generating embeddings that are combined with structured data to form an enhanced feature representation.

3.6.4. Dimensionality Reduction

To mitigate feature redundancy and improve computational efficiency, PCA is applied to project augmented features into a lower-dimensional space while retaining key information.

These preprocessing steps refine data representation, balancing structure and semantics. Figure 4 illustrates PCA results and normalized feature distributions.

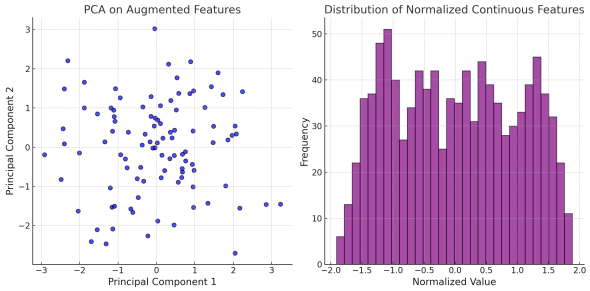

Figure 4. PCA on Augmented Features and Distribution of Normalized Continuous Features.

3.7. Evaluation Metrics

The ALIMN framework is evaluated using standard metrics.

Accuracy measures overall correctness:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{15}$$

Precision quantifies the correctness of positive predictions:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{16}$$

Recall assesses sensitivity to positive cases:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{17}$$

F1-score balances precision and recall:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{18}$$

ROC AUC evaluates class separability:

$$\text{AUC} = \int_0^1 \text{TPR}\left(\text{FPR}^{-1}(x)\right) dx. \tag{19}$$

 doi:10.20944/preprints202506.0994.v1

4. Experiment Results

We conducted extensive experiments to evaluate the performance of the ALIMN framework and its variants, as well as to compare it with several mainstream models. Table 1 summarizes the performance of these models on a test dataset using the evaluation metrics defined previously. The changes in model training indicators are shown in Figure 5.

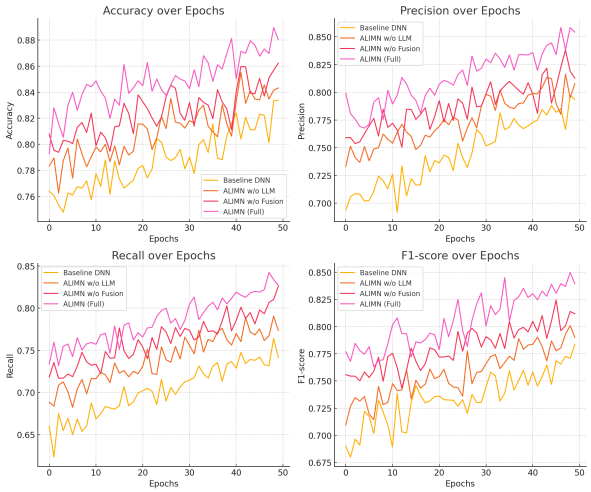

Figure 5. Model indicator change chart.

Table 1. Ablation Study: Performance Comparison of ALIMN Variants.

Model	Accuracy	Precision	Recall	F1-score	ROC AUC
Baseline DNN	0.82	0.79	0.75	0.77	0.84
ALIMN w/o LLM	0.84	0.81	0.78	0.79	0.86
ALIMN w/o Fusion	0.85	0.82	0.80	0.81	0.87
ALIMN (Full)	0.88	0.85	0.83	0.84	0.90

Table 2 presents the performance comparison between ALIMN and these mainstream models.

Table 2. Comparison of ALIMN with Mainstream Models.

Model	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression (LR)	0.79	0.75	0.73	0.74	0.81
Random Forest (RF)	0.83	0.80	0.78	0.79	0.85
XGBoost (XGB)	0.84	0.81	0.79	0.80	0.86
Support Vector Machine (SVM)	0.80	0.77	0.74	0.75	0.82
ALIMN (Full)	0.88	0.85	0.83	0.84	0.90

5. Conclusions

In conclusion, the proposed ALIMN framework, through its innovative integration of LLM-based semantic feature extraction, contextual fusion, and a multi-branch ensemble architecture, achieves superior predictive performance compared to conventional deep learning and traditional machine learning models. The ablation studies and comparisons with mainstream models validate its effectiveness and robustness, highlighting its potential as a promising solution for complex predictive tasks in the financial domain.

References

1. F. M. Talaat, A. Aljadani, B. Alharthi, M. A. Farsi, M. Badawy, and M. Elhosseini, "A mathematical model for customer segmentation leveraging deep learning, explainable ai, and rfm analysis in targeted marketing," *Mathematics*, vol. 11, no. 18, p. 3930, 2023.

 doi:10.20944/preprints202506.0994.v1

2. C. S. Potluri, G. S. Rao, L. M. Kumar, K. G. Allo, Y. Awoke, and A. A. Seman, "Machine learning-based customer segmentation and personalised marketing in financial services," in *2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE)*. IEEE, 2024, pp. 1570–1574.
3. T. Jin, "Attention-based temporal convolutional networks and reinforcement learning for supply chain delay prediction and inventory optimization," *Preprints*, January 2025. [Online]. Available: https://doi.org/10.20944/preprints202501.1543.v1
4. T. N. Pandey, N. K. SV, M. Amrutha, B. B. Dash, and S. S. Patra, "Experimental analysis on banking customer segmentation using machine learning techniques," in *2023 Global Conference on Information Technologies and Communications (GCITC)*. IEEE, 2023, pp. 1–6.
5. J. Tang, W. Qian, L. Song, X. Dong, L. Li, and X. Bai, "Optimal boxes: boosting end-to-end scene text recognition by adjusting annotated bounding boxes via reinforcement learning," in *European Conference on Computer Vision*. Springer, 2022, pp. 233–248.
6. H. Feng, Q. Liu, H. Liu, J. Tang, W. Zhou, H. Li, and C. Huang, "Docpedia: Unleashing the power of large multimodal model in the frequency domain for versatile document understanding," *Science China Information Sciences*, vol. 67, no. 12, pp. 1–14, 2024.
7. J. Tang, S. Qiao, B. Cui, Y. Ma, S. Zhang, and D. Kanoulas, "You can even annotate text with voice: Transcription-only-supervised text spotting," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 4154–4163.
8. B. Lu, H.-C. Dan, Y. Zhang, and Z. Huang, "Journey into automation: Image-derived pavement texture extraction and evaluation," *arXiv preprint arXiv:2501.02414*, 2025.
9. J. Tang, W. Zhang, H. Liu, M. Yang, B. Jiang, G. Hu, and X. Bai, "Few could be better than all: Feature sampling and grouping for scene text detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4563–4572.
10. Y. Liu, J. Zhang, D. Peng, M. Huang, X. Wang, J. Tang, C. Huang, D. Lin, C. Shen, X. Bai *et al.*, "Spts v2: single-point scene text spotting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 12, pp. 15 665–15 679, 2023.
11. Z. Zhao, J. Tang, B. Wu, C. Lin, S. Wei, H. Liu, X. Tan, Z. Zhang, C. Huang, and Y. Xie, "Harmonizing visual text comprehension and generation," *arXiv preprint arXiv:2407.16364*, 2024.
12. A. Julian and S. Hariprasath, "Optimizing customer segmentation through machine learning," in *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, vol. 5. IEEE, 2024, pp. 413–416.
13. H.-C. Dan, B. Lu, and M. Li, "Evaluation of asphalt pavement texture using multiview stereo reconstruction based on deep learning," *Construction and Building Materials*, vol. 412, p. 134837, 2024.
14. T. Jin, "Optimizing retail sales forecasting through a pso-enhanced ensemble model integrating lightgbm, xgboost, and deep neural networks," *Preprints*, January 2025. [Online]. Available: https://doi.org/10.20944/preprints202501.1604.v1
15. J. Tang, Q. Liu, Y. Ye, J. Lu, S. Wei, C. Lin, W. Li, M. F. F. B. Mahmood, H. Feng, Z. Zhao *et al.*, "Mtvqa: Benchmarking multilingual text-centric visual question answering," *arXiv preprint arXiv:2405.11985*, 2024.
16. H.-C. Dan, Z. Huang, B. Lu, and M. Li, "Image-driven prediction system: Automatic extraction of aggregate gradation of pavement core samples integrating deep learning and interactive image processing framework," *Construction and Building Materials*, vol. 453, p. 139056, 2024.
17. J. Tang, C. Lin, Z. Zhao, S. Wei, B. Wu, Q. Liu, H. Feng, Y. Li, S. Wang, L. Liao *et al.*, "Textsquare: Scaling up text-centric visual instruction tuning," *arXiv preprint arXiv:2404.12803*, 2024.
18. Z. Zhao, J. Tang, C. Lin, B. Wu, C. Huang, H. Liu, X. Tan, Z. Zhang, and Y. Xie, "Multi-modal in-context learning makes an ego-evolving scene text recognizer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 567–15 576.
19. J. Tang, W. Du, B. Wang, W. Zhou, S. Mei, T. Xue, X. Xu, and H. Zhang, "Character recognition competition for street view shop signs," *National Science Review*, vol. 10, no. 6, p. nwad141, 2023.