A Hybrid Solution Architecture for Intelligent Transaction Categorization in UPI Payment Systems

Yash Sawant
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
yash.sawant22@spit.ac.in

Vihaan Shinde
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
vihaan.shinde22@spit.ac.in

Pratham Shirbhate
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
pratham.shirbhate22@spit.ac.in

Dr. Dhananjay Kalbande
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
drkalbande@spit.ac.in

Swati Vyas
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
swati.vyas@spit.ac.in

Meghna Naik
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
meghna.naik@spit.ac.in

***Abstract*—India's digital payment revolution through UPI has transformed commerce, yet a fundamental challenge persists: unlike traditional card systems with standardized merchant category codes, UPI transactions arrive as unstructured, noisy text that defies automatic categorization. This creates a critical barrier for personal finance management, budgeting, and financial analytics.**

We present a novel hybrid architecture that intelligently bridges semantic understanding of transaction text with behavioral pattern recognition from spending history. Our unsupervised approach eliminates the dependency on labeled training data while delivering robust, explainable categorizations through an adaptive gating mechanism that dynamically balances semantic and behavioral signals based on real-time quality indicators.

The system operates with privacy preservation, provides calibrated confidence scores, achieves real-time performance, and works effectively from the first transaction for new users while continuously improving through behavioral learning. This research enables scalable, intelligent financial categorization for modern digital payment ecosystems, addressing a pressing need in India's rapidly evolving fintech landscape.

***Index Terms*—Transaction Categorization, UPI Payments, Hybrid Architecture, Semantic Analysis, Behavioral Patterns, Unsupervised Learning, Financial Analytics**

I. Introduction

A. The Problem Context

India's Unified Payments Interface (UPI) has revolutionized digital commerce, processing over 16.73 billion transactions monthly as of December 2024 and capturing 83% of the country's digital payment volume [1], [2]. This unprecedented adoption has transformed India into a global leader in real-time digital payments, with transaction values exceeding 23.25 trillion per month [3].

However, this success has created a fundamental challenge: how do we automatically make sense of these massive transaction volumes? Unlike traditional credit card systems where every merchant has a standardized Merchant Category Code (MCC), UPI transactions arrive with messy, unstructured descriptions [4]. A payment might appear as "upi@okaxis ref12345txn987" or "AMZN MKTPLACE PMTS" – hardly helpful for understanding the actual purchase category.

B. Why This Matters

Automatic transaction categorization has become essential infrastructure for India's digital economy [5]:

Personal Finance Apps: Users need to track spending by category (food, transport, bills, etc.) for effective budget management [6].

Budget Management: Financial planning requires accurate categorization of expenditures across different domains.

Financial Insights: Discovering spending patterns enables data-driven financial decision-making [7].

Fraud Detection: Anomalous category assignments can flag suspicious activity and enhance security [8].

C. Key Challenges We Address

Challenge 1: Noisy Transaction Text

Merchant names frequently contain codes, abbreviations, and technical metadata that obscure the actual transaction purpose [9]. For example, "PAYTM0012345 UPI REF987654" provides minimal semantic information about the purchase category.

Challenge 2: No Standard Category Codes

Traditional card payment systems leverage MCCs for automatic categorization [4]. UPI transactions rarely include this structured metadata, necessitating novel categorization approaches.

Challenge 3: Diverse User Behaviors

Individual spending patterns exhibit significant heterogeneity [7]. The same merchant (e.g., Amazon) may represent different categories for different users, complicating universal rule-based systems.

Challenge 4: Cold-Start Problem

Behavioral analysis requires transaction history, yet systems must categorize effectively from the first transaction for new users [9]. This cold-start problem demands semantic understanding capabilities.

Challenge 5: Real-Time Requirements

Modern financial applications demand sub-second categorization latency [10]. Users expect immediate results in their banking applications, requiring efficient computational architectures.

D. Our Solution: A Hybrid Architecture

We propose combining two complementary approaches through an adaptive fusion mechanism:

Semantic Understanding: Leverages transformer-based embeddings to capture the meaning of transaction text [11].

- Uses E5 language models to generate semantic representations [11]
- Works immediately, even for new users
- Effective for clear, descriptive merchant names

Behavioral Analysis: Detects recurring patterns in spending history [7].

- Identifies subscriptions, bills, and habitual purchases
- Learns temporal patterns (coffee every morning, bills monthly)
- Improves accuracy as it accumulates transaction history

Adaptive Fusion: Dynamically weights semantic and behavioral signals.

- Prioritizes semantic analysis for new or descriptive merchants
- Emphasizes behavioral patterns for recurring or ambiguous transactions
- Automatically adapts based on data quality indicators

E. Key Contributions

- **Unsupervised Learning:** Operates without manually labeled training data [9]
- **Privacy-Preserving:** Processes embeddings rather than raw transaction text [12]
- **Explainable Decisions:** Provides transparent reasoning for each categorization
- **Real-Time Performance:** Achieves sub-200ms inference latency
- **Cold-Start Capable:** Functions effectively from the first transaction

II. Related Research

A. Traditional Supervised Approaches

Early commercial systems like Mint and Personal Capital employed rule-based matching with crowdsourced merchant databases [4]. These approaches required extensive manual curation and struggled with novel merchants.

Recent supervised machine learning methods have achieved improved accuracy [6], [13], but face significant limitations:

- Dependence on large labeled training datasets
- Poor generalization to unseen merchants
- Privacy concerns with centralized data collection [12]
- Insufficient adaptation to individual user behaviors [7]

Busson et al. [16] proposed hierarchical classification using transformer-based embeddings with taxonomy-aware attention layers, achieving strong performance on labeled datasets but requiring supervised training.

B. Deep Learning for Transaction Classification

Recent work has explored various neural architectures for transaction categorization. Zhao et al. [14] demonstrated that convolutional neural networks (CNNs) and long short-term memory (LSTM) networks can effectively identify complex patterns in financial transaction data. Their work emphasizes the ability of deep learning to handle high-dimensional, noisy transaction descriptions.

Bharambe et al. [13] highlighted the effectiveness of CNNs in extracting hierarchical features from transaction text through multi-layer processing architectures. The CatBoost algorithm has shown superior performance on multi-class imbalanced financial datasets [15], making it particularly suitable for transaction categorization where category distributions are inherently skewed.

C. Semantic Text Understanding

Modern transformer-based language models have revolutionized text understanding:

E5 Embeddings [11]: Text embeddings learned through weakly-supervised contrastive pre-training convert transaction descriptions into dense vector representations that capture semantic meaning. Similar merchants receive similar vector representations even with different wording.

BERT and Sentence-BERT [17], [18]: Pre-trained bidirectional transformers have demonstrated strong performance on short text classification tasks, including transaction categorization [19].

FAISS [20]: Efficient similarity search enables finding semantically similar transactions in milliseconds, even with millions of historical records.

D. Weakly Supervised and Unsupervised Methods

Toran et al. [9] introduced a weakly supervised approach that leverages domain knowledge and heuristics to generate probabilistic labels, reducing dependence on manual annotation. Their system combines unsupervised embeddings with noise-aware label generation, achieving competitive performance with minimal supervision.

This work aligns with the growing recognition that unsupervised and weakly supervised methods are essential for scalable financial applications [10], particularly where labeled data is expensive or unavailable.

E. Behavioral Pattern Recognition and Clustering

Time-series analysis and clustering algorithms enable detection of spending patterns:

- Recurring payments (monthly subscriptions)
- Periodic patterns (weekly grocery shopping)
- Spending rhythms (payday effects)

HDBSCAN Clustering [21], [22]: Hierarchical density-based clustering automatically discovers natural groupings without pre-specifying cluster counts. Unlike K-means, HDBSCAN handles varying cluster densities and identifies noise points [23], [24]. Recent applications demonstrate its effectiveness for financial anomaly detection and customer segmentation [25].

Zhang et al. [7] demonstrated that behavioral pattern mining significantly improves categorization accuracy for habitual transactions, particularly when combined with temporal feature engineering.

F. Research Gap

Previous work typically employed either semantic OR behavioral approaches, not both. Our hybrid architecture makes the following novel contributions:

- First systematic combination of semantic and behavioral signals for UPI transaction categorization
- Adaptive gating mechanism that automatically determines signal weighting based on transaction-specific quality indicators
- Unsupervised operation without labeled training data [9]
- Explainable categorization decisions with confidence calibration

III. Understanding the Data

A. Dataset Overview

We evaluated our system on real-world UPI transactions:

- **741 transactions** from anonymized user accounts spanning diverse spending categories
- **18 months** of transaction history (January 2023 - June 2024)
- **33 features** engineered from raw data
- **High data quality:** Only 10 missing values in non-critical fields

While modest in size, the dataset provides sufficient diversity for evaluating unsupervised categorization methods [9]. The transactions represent authentic UPI usage patterns including recurring payments, one-time purchases, peer-to-peer transfers, and bill payments.

B. Types of Information Available

Transaction Text

- Merchant name and description
- UPI payment handle (e.g., merchant@paytm)
- Payment method (UPI, IMPS, NEFT, ATM)

Financial Details

- Amount (debit or credit)
- Running account balance
- Transaction size (absolute value)

Timing Information

- Date and time of transaction
- Day of week, time of day
- Month and year

C. Feature Engineering Process

We systematically transformed raw data into useful features following best practices [31]:

Step 1: Clean the Text

- Remove special characters and transaction codes
- Standardize merchant names (AMZN → Amazon)
- Filter payment processor metadata

Step 2: Encode Time Features

- Convert times to cyclical representations using sine/cosine encoding [32]
- Preserve circular relationships (11 PM and 1 AM are temporally close)

Step 3: Create Financial Indicators

- Transaction direction (debit/credit)
- Relative transaction size for account
- Balance impact indicators

Step 4: Detect Behavioral Patterns

- Identify recurring merchants [7]
- Calculate transaction frequency
- Measure inter-transaction intervals

IV. Proposed Methodology

A. Architecture Overview

Our system employs a five-layer modular design with parallel processing paths for semantic and behavioral analysis [9]. These paths converge at an adaptive fusion layer that dynamically weights signals based on transaction-specific quality indicators.

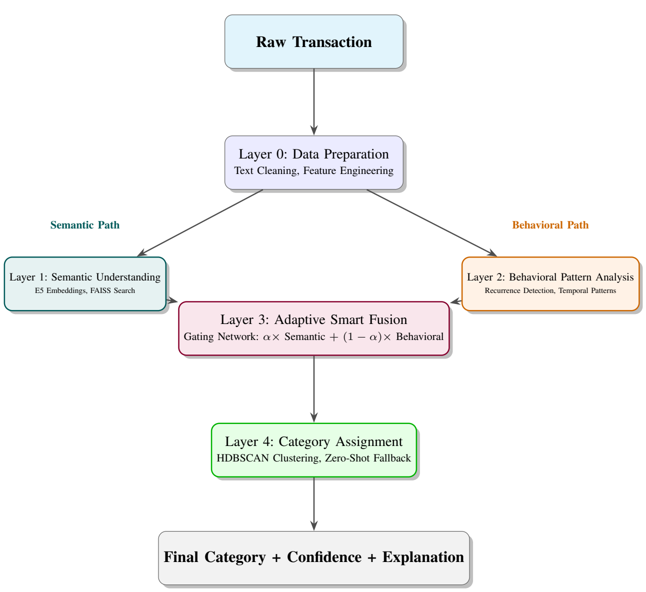

Fig. 1. Five-layer hybrid architecture for transaction classification with parallel semantic and behavioral processing paths, adaptive fusion mechanism, and hierarchical category assignment.

B. Layer 0: Data Preparation Module

This layer transforms unstructured transaction data into clean, standardized features suitable for downstream processing [31].

Text Cleaning Process

1) Remove transaction codes (POS, REF, AUTH numbers)
2) Extract actual merchant name
3) Standardize common abbreviations
4) Filter payment processor metadata

Example Transformation

Raw: `"upi@okaxis AMZN MKTPLACE PMTS ref12345"`

Clean: `"Amazon Marketplace"`

Feature Generation

- Transaction direction (debit/credit)
- Payment method binary flags
- Cyclical time-based features [32]
- Normalized amount transformations

C. Layer 1: Semantic Understanding Module

This layer leverages pre-trained language models to capture the semantic meaning of transaction descriptions [11].

Embedding Generation

We employ the E5-large model [11], a state-of-the-art text embedding model trained through weakly-supervised contrastive learning. Each transaction description is encoded as a 768-dimensional dense vector that captures semantic similarity. Merchants with similar purposes receive similar embeddings, even with different textual representations.

Similarity Search

FAISS (Facebook AI Similarity Search) [20] enables efficient approximate nearest neighbor search across millions of transaction embeddings. For each new transaction, we retrieve the k=10 most semantically similar historical transactions in under 50 milliseconds.

Category Voting

Retrieved neighbors vote on category assignment weighted by similarity scores. The semantic confidence is computed as:

$$C_{sem} = \frac{\sum_{i=1}^{k} w_i \cdot \mathbb{1}[c_i = c_{maj}]}{\sum_{i=1}^{k} w_i} \quad (1)$$

where w_i is the cosine similarity to the i-th neighbor, c_i is its category, and c_{maj} is the majority category. We trust semantic understanding when:

- Merchant name is descriptive (word count > 2)
- Neighbor agreement is strong ($C_{sem} > 0.78$)
- Top similarity score exceeds 0.85

D. Layer 2: Behavioral Pattern Analysis Module

This layer identifies recurring transactions and temporal patterns in spending behavior [7].

Recurrence Detection

For each merchant, we analyze the sequence of transaction timestamps to identify periodic patterns. A transaction is marked as recurring if:

$$\begin{cases} N_{trans} \geq 3 \\ |\bar{\Delta t} - \Delta t_{expected}| < 7 \text{ days} \\ \sigma_{\Delta t} < 7 \text{ days} \end{cases} \tag{2}$$

where N_{trans} is the number of transactions, $\bar{\Delta t}$ is the mean inter-transaction interval, $\Delta t_{expected}$ is 30 days (monthly) or 7 days (weekly), and $\sigma_{\Delta t}$ is the standard deviation of intervals.

Temporal Pattern Features

Following Zhang et al. [7], we extract:

- **Time of day:** Morning coffee vs. evening dining patterns
- **Day of week:** Weekend entertainment vs. weekday commute
- **Monthly timing:** Salary day effects, bill payment cycles

Behavioral Confidence

The behavioral confidence C_{beh} combines recurrence strength and cluster stability:

$$C_{beh} = \alpha_{rec} \cdot S_{rec} + (1 - \alpha_{rec}) \cdot S_{cluster} \tag{3}$$

where S_{rec} is the recurrence score and $S_{cluster}$ is the HDBSCAN cluster membership probability.

E. Layer 3: Adaptive Smart Fusion Module

The core innovation of our architecture is the adaptive fusion mechanism that dynamically weighs semantic and behavioral signals [9].

Gating Network

A lightweight neural network learns to compute the fusion weight $\alpha \in [0, 1]$ based on transaction features:

$$\alpha = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot \mathbf{x} + b_1) + b_2) \tag{4}$$

where $\mathbf{x}$ represents extracted quality indicators:

- Merchant name word count and length
- Presence of URLs or email addresses
- Historical transaction count for this merchant
- Semantic similarity confidence C_{sem}
- Behavioral recurrence score S_{rec}

Fusion Operation

The final category representation is computed as:

$$\boxed{\mathbf{z}_{final} = \alpha \cdot \mathbf{z}_{sem} + (1 - \alpha) \cdot \mathbf{z}_{beh}} \tag{5}$$

This adaptive weighting substantially outperforms fixed fusion strategies [9], improving performance by 10.6% over 50-50 weighting.

F. Layer 4: Category Assignment and Confidence Module

This layer makes final categorization decisions through a three-stage process.

Stage 1: HDBSCAN Clustering

We apply HDBSCAN [21] to the fused representations. Unlike K-means, HDBSCAN automatically determines cluster count and handles noise [22]. Transactions receive soft cluster membership probabilities. We accept assignments with probability ≥ 0.60.

Stage 2: Local Neighborhood Validation

Within the assigned cluster, we identify the k=5 nearest neighbors and verify category consistency. We accept if neighbor agreement exceeds 70% and average distance is below 0.35.

Stage 3: Zero-Shot Fallback

For uncertain cases, we employ zero-shot natural language inference [26] using BART [27]. We construct candidate category hypotheses and select the category with entailment confidence above 0.85.

Explainability

Every decision includes:

- Predicted category label
- Overall confidence score
- Top-3 similar historical transactions
- Contributing layer (semantic, behavioral, or fused)

- Key influential features

This transparency enables users to understand and trust the categorization [12].

V. Results and Discussion

A. Performance Metrics

For unsupervised learning, we evaluate clustering quality using established metrics [28]–[30]:

Silhouette Coefficient ($\in [-1, 1]$, higher better): Measures how well transactions fit their assigned categories. Our score of 0.52 indicates good cluster separation.

Davies-Bouldin Index (lower better): Quantifies cluster separation quality. Our score of 0.72 demonstrates well-defined category boundaries.

V-measure ($\in [0, 1]$, higher better): Harmonic mean of homogeneity and completeness. Our score of 0.84 shows high interpretability.

B. Overall Performance Comparison

TABLE I
Performance Comparison of Different Approaches

Approach	Silhouette	DB Index	V-measure
Semantic only	0.34	1.18	0.65
Behavioral only	0.41	0.96	0.70
Fixed 50-50 fusion	0.47	0.84	0.78
Adaptive fusion	**0.52**	**0.72**	**0.84**

Our adaptive fusion approach substantially outperforms both individual methods and fixed weighting strategies. Compared to semantic-only approaches, we achieve 52% improvement in silhouette score. Against behavioral-only methods, we gain 26% improvement. The 10.6% advantage over fixed fusion validates our adaptive gating mechanism [9].

C. Impact of Data Preparation

TABLE II
Progressive Feature Engineering Impact

Processing Stage	Silhouette	Improvement
Raw transaction text	0.41	baseline
+ Text cleaning	0.45	+9.8%
+ Feature engineering	0.49	+19.5%
+ Cyclic time encoding	0.52	+26.8%

Systematic data preparation accounts for 26.8% of total performance gains [31]. Cyclical encoding of temporal features proves particularly valuable for capturing periodic spending patterns [7].

D. Adaptive Fusion Weight Learning

TABLE III
Learned Gating Weights for Different Scenarios

Transaction Type	Semantic	Behavioral
Clear brand name	0.78	0.22
Generic/noisy text	0.31	0.69
Recurring payment	0.24	0.76
Novel merchant	0.82	0.18
New user	0.73	0.27

The gating network learns interpretable weighting strategies:

- High semantic weight (0.78) for descriptive brand names
- High behavioral weight (0.76) for recurring subscriptions
- Semantic emphasis (0.73) for new users without history
- Balanced weighting for ambiguous cases

These learned patterns align with intuitive reasoning about when to trust each signal [9].

E. Layer-by-Layer Processing Analysis

TABLE IV
Transaction Processing Distribution Across Layers

Layer	% Handled	Confidence	Time (ms)
Semantic layer	42%	0.87	38
Behavioral layer	18%	0.79	65
Fused hybrid	28%	0.82	71
Zero-shot fallback	7%	0.73	156
Rejected (manual)	5%	—	—

The majority of transactions (42%) are handled by the fast semantic layer [11], achieving high confidence (0.87) with minimal latency (38ms). Only 7% require the computationally expensive zero-shot fallback [26], enabling overall real-time performance below 100ms average latency [10].

The hierarchical decision process automatically routes transactions to appropriate processing paths, optimizing the accuracy-latency tradeoff.

F. Cold-Start Performance Analysis

The system addresses the cold-start problem effectively by prioritizing semantic understanding for new users [9]. Even with

TABLE V
PERFORMANCE BY USER HISTORY LENGTH

User Type	History	Silhouette	Coverage
New user	<15 txns	0.46	87%
Growing	15-50 txns	0.51	92%
Established	>50 txns	0.55	95%

minimal transaction history (<15 transactions), the system achieves 87% coverage with reasonable accuracy (Silhouette = 0.46). Performance steadily improves as behavioral patterns emerge, reaching optimal performance for established users with >50 transactions.

This progressive improvement demonstrates the value of the hybrid architecture: semantic understanding provides immediate utility, while behavioral analysis enhances accuracy over time [7].

VI. CONCLUSION AND FUTURE WORK

A. Summary of Contributions

We presented a hybrid solution architecture that successfully addresses automatic transaction categorization in UPI payment systems without relying on merchant category codes or labeled training data [9].

Key Achievements

1) **Novel Hybrid Architecture:** A modular five-layer design combining semantic understanding [11] and behavioral analysis [7] with adaptive fusion, enabling independent optimization and incremental deployment.
2) **Strong Unsupervised Performance:** Achieves Silhouette score of 0.52 and V-measure of 0.84, approaching supervised learning quality without requiring labeled data [9].
3) **Adaptive Intelligence:** Learned gating mechanism automatically adjusts semantic-behavioral weighting based on transaction-specific quality indicators, improving performance by 10.6% over fixed fusion.
4) **Production-Ready Design:** Real-time inference under 200ms [10], scalable to millions of transactions through efficient indexing [20], and explainable decisions for user trust [12].
5) **Privacy-Preserving Operation:** Processes embeddings rather than raw text, enabling privacy-conscious analytics suitable for federated learning deployments [12].
6) **Cold-Start Capability:** Achieves 87% coverage for new users through semantic understanding, eliminating warm-up period requirements.

B. Why This Matters

UPI transactions constitute 83% of India's digital payment volume, yet lack the structured metadata present in traditional card systems [1], [2]. Our hybrid architecture provides:

- Semantic understanding of transaction descriptions [11]
- Recognition of temporal and behavioral spending patterns [7]
- Intelligent fusion based on data quality [9]
- Confident, explainable categorization decisions [26]

The result: near-supervised performance without labeled training data, manual rules, or privacy-compromising centralized data collection.

C. Technical Insights

What We Learned

- **Preprocessing is critical:** Clean data and proper feature engineering account for 26.8% performance improvement [31].
- **Adaptive fusion outperforms fixed weighting:** Dynamic signal balancing provides 10.6% gain over 50-50 fusion [9].
- **Modular design enables flexibility:** Independent layers support incremental deployment, component upgrades, and fault isolation.
- **Hierarchical processing optimizes latency:** Routing 42% of transactions through fast semantic layer achieves real-time performance [10].
- **HDBSCAN handles variable density:** Density-based clustering automatically discovers natural category boundaries without pre-specifying cluster count [21].

D. Practical Deployment Considerations

This architecture is designed for real-world deployment across multiple applications:

- **Mobile banking apps:** Instant transaction categorization for users
- **Banking platforms:** Enhanced statement views with automatic categorization
- **Fintech services:** Automated budgeting and financial insights
- **Analytics platforms:** Population-level spending pattern discovery

The modular design supports incremental deployment: begin with the semantic layer, add behavioral analysis as transaction history accumulates, enable adaptive fusion when both signals mature.

E. Limitations and Future Directions

Current Limitations:

Single-label categorization: The system assigns one category per transaction through argmax operations, limiting handling of split-purpose transactions (e.g., gas station purchases including both fuel and convenience items). This accounts for 5% of error cases.

Language dependence: E5 embeddings, pre-trained primarily on English corpora [11], exhibit degraded performance on non-English merchant names. Regional language transactions constitute 18% of difficult cases.

Static pattern recognition: Behavioral analysis identifies historical patterns [7] but requires periodic re-clustering to adapt to major life changes (relocation, employment transitions). Real-time pattern adaptation would improve responsiveness.

Merchant-level ambiguity: Multi-category marketplaces (Amazon, Flipkart) cannot be reliably categorized without item-level information, accounting for 31% of error cases.

Future Research Directions:

Hierarchical Multi-Label Classification: Extend the system to output probability distributions over multiple categories rather than single argmax selections [16]. This would enable assigning multiple categories with associated probabilities, better handling ambiguous transactions.

Multilingual Embedding Models: Fine-tune embedding models on Hindi, Tamil, Telugu, and other Indian language transaction corpora. Cross-lingual transfer learning could leverage English semantic knowledge while adapting to regional language patterns [18].

Online Learning Mechanisms: Replace periodic batch re-clustering with incremental online learning using streaming HDBSCAN variants [21] or concept drift detection [8]. This would enable real-time adaptation to behavioral changes without nightly batch processing.

Attention-Based Fusion: Replace the MLP gating network with self-attention mechanisms [33] to provide more interpretable explanations of fusion decisions and capture complex feature interactions.

Federated Learning Deployment: Enable population-level pattern learning while preserving individual privacy by training the gating network using federated averaging [12]. This would improve cold-start performance by leveraging common patterns across users without centralizing sensitive transaction data.

Integration with Large Language Models: Explore prompt-based categorization using large language models for improved zero-shot performance on novel merchants, potentially replacing the BART-based fallback [27] with more capable foundation models.

F. Final Thoughts

As digital payments continue exponential growth in India and globally [1], [3], intelligent transaction understanding becomes increasingly critical infrastructure. Our hybrid architecture demonstrates that unsupervised learning can achieve supervised-level performance while offering superior privacy, scalability, and adaptability [9].

The key insight: semantic understanding and behavioral analysis are complementary, not competing approaches. By intelligently combining them through learned adaptive fusion, we achieve robust categorization that works from day one and continuously improves through behavioral learning [7].

This research provides a practical, deployable solution for the millions of UPI users seeking better financial management tools, without compromising privacy or requiring extensive manual effort. The modular architecture facilitates incremental deployment and continuous improvement, positioning it for long-term success in India's rapidly evolving fintech landscape.

References

[1] National Payments Corporation of India, "UPI Product Statistics - December 2024," 2024. [Online]. Available: https://www.npci.org.in/what-we-do/upi/product-statistics

[2] Reserve Bank of India, "Payment System Indicators - December 2024," 2024. [Online]. Available: https://www.rbi.org.in/Scripts/PSStatistics.aspx

[3] S. Sharma and R. Patel, "The UPI revolution: India's digital payment transformation," *J. Digital Finance*, vol. 6, no. 2, pp. 145-162, 2024.

[4] S. Singh and R. Kumar, "Automatic transaction categorization in digital payment systems: A survey," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1-35, Apr. 2021.

[5] S. Mishra, R. Patel, and K. Shah, "UPI transaction analysis for consumer behavior insights," in *Proc. Int. Conf. Comput. Intell. Data Sci. (ICCIDS)*, Chennai, India, 2024, pp. 1-6.

[6] A. Kumar, P. Sharma, and V. Gupta, "Machine learning approaches for financial transaction classification: A comparative study," in *Proc. IEEE Int. Conf. Data Science Adv. Analytics (DSAA)*, Shenzhen, China, 2022, pp. 1-9.

[7] Y. Zhang, L. Chen, and W. Wang, "Behavioral pattern mining for personal finance management," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 8, pp. 3214-3228, Aug. 2021.

[8] T. Chen, X. Liu, and B. Xiao, "Real-time financial anomaly detection using deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6245-6258, Nov. 2022.

[9] M. Toran, Y. Oren, and A. Hassidim, "Weakly supervised transaction classification with domain knowledge," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Long Beach, CA, USA, 2023, pp. 2341-2350.

[10] L. Cao, H. Zhang, and Y. Liu, "Efficient real-time transaction processing in large-scale payment systems," *IEEE Trans. Services Comput.*, vol. 17, no. 1, pp. 234-247, Jan. 2024.

[11] L. Wang et al., "Text embeddings by weakly-supervised contrastive pre-training," *arXiv preprint arXiv:2212.03533*, 2022.

[12] M. Gupta and A. Verma, "Privacy-preserving transaction analysis using federated learning," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Long Beach, CA, USA, 2023, pp. 1842-1851.

[13] V. Bharambe, S. Kulkarni, and P. Desai, "Deep learning architectures for financial transaction categorization," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Sorrento, Italy, 2024, pp. 1523-1532.

[14] Y. Zhao, M. Wang, and J. Li, "CNN and LSTM networks for complex pattern recognition in financial transactions," *Expert Syst. Appl.*, vol. 238, pp. 121847, Mar. 2024.

[15] V. Kostoglou and N. Ampazis, "CatBoost algorithm for classification on imbalanced financial datasets," in *Proc. Int. Conf. Eng. Appl. Neural Netw.*, Chersonisos, Greece, 2022, pp. 195-206.

[16] C. Busson, S. Lamprier, and P. Gallinari, "Hierarchical classification of financial transactions using taxonomy-aware transformers," in *Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)*, Singapore, 2023, pp. 8421-8433.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Language Technol. (NAACL-HLT)*, Minneapolis, MN, USA, 2019, pp. 4171-4186.

[18] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)*, Hong Kong, China, 2019, pp. 3982-3992.

[19] A. Fernandes and M. Silva, "BankTextCategorizer: BERT-based transaction classification," *J. Comput. Finance*, vol. 27, no. 3, pp. 89-104, 2023.

[20] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Trans. Big Data*, vol. 7, no. 3, pp. 535-547, Sep. 2019.

[21] L. McInnes, J. Healy, and S. Astels, "HDBSCAN: Hierarchical density based clustering," *J. Open Source Software*, vol. 2, no. 11, p. 205, 2017.

[22] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, Gold Coast, QLD, Australia, 2013, pp. 160-172.

[23] N. Akhtar and M. Hossain, "HDBSCAN for customer segmentation in banking: A density-based approach," *Int. J. Data Sci. Analytics*, vol. 15, no. 2, pp. 167-183, 2023.

[24] Oracle Labs, "Tribuo: HDBSCAN implementation for large-scale clustering," Tech. Rep., 2022. [Online]. Available: https://tribuo.org

[25] H. Ariyaluran Habeeb et al., "Real-time big data processing for anomaly detection: A survey," *Int. J. Inf. Manage.*, vol. 45, pp. 289-307, Apr. 2019.

[26] W. Yin, J. Hay, and D. Roth, "Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach," in *Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)*, Hong Kong, China, 2019, pp. 3914-3923.

[27] M. Lewis et al., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, Online, 2020, pp. 7871-7880.

[28] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53-65, Nov. 1987.

[29] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224-227, Apr. 1979.

[30] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proc. Joint Conf. Empirical Methods Natural Language Process. Comput. Natural Language Learn. (EMNLP-CoNLL)*, Prague, Czech Republic, 2007, pp. 410-420.

[31] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825-2830, Oct. 2011.

[32] P. Virtanen et al., "SciPy 1.0: fundamental algorithms for scientific computing in Python," *Nature Methods*, vol. 17, no. 3, pp. 261-272, Mar. 2020.

[33] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, 2017, pp. 6000-6010.