



FInsightAI: An AI-Powered Predictive Financial Intelligence Mobile Application — A Comprehensive Literature Review

Siddhant Wankhade¹, Joel Patankar², Mohammad Aatir Fawad Ansari³, Prof. Shivani Jagtap⁴

¹ Dept. of AIML, GNIT, RTMNU, Nagpur, India

² Dept. of AIML, GNIT, Nagpur, India

³ Dept. of AIML, GNIT, Nagpur, India

⁴ Dept. of AIML, GNIT (Head of Department & Project Supervisor), Nagpur, India

Corresponding Author Email: siddhantw291@gmail.com

Abstract—

How to Cite this Article:

Wankhade, S., Patankar, J. & Ansari, M. A. F. (2026). FInsightAI: An AI-Powered Predictive Financial Intelligence Mobile Application — A Comprehensive Literature Review. International Journal of Creative and Open Research in Engineering and Management, 2(9), 1-9. <https://doi.org/10.55041/ijcope.v2i9.173>

License:

This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.



<https://doi.org/10.55041/ijcope.v2i9.173>

Traditional Personal Finance Management (PFM) applications are passive tools that rely on manual data entry, produce static reports, and offer no forward-looking financial intelligence. A newer wave of AI-powered tools introduced Large Language Model (LLM) wrappers, but these suffer from a critical flaw: mathematical hallucinations, where the model generates confident but numerically incorrect financial figures. This paper presents FInsightAI, a native Android mobile application built with React Native and Firebase that directly addresses both problems. FInsightAI automatically captures transaction data via Android SMS broadcast listeners, eliminating all manual entry. It processes this data through a three-tier auto-categorization engine and a deterministic data science backend comprising Holt-Winters Triple Exponential Smoothing for 12-to-36 month cash flow forecasting, Unsupervised Dynamic Z-Score anomaly detection, and Gaussian Kernel Density Estimation (KDE) for peer cohort benchmarking. A Zod schema validation layer ensures the OpenAI GPT-4o-mini advisory engine receives only verified numerical facts, guaranteeing zero mathematical hallucinations. A gap analysis of 2023-2026 literature confirms that no prior work unifies all these components in a single production-grade native mobile framework.

Keywords— Anomaly

Detection; Cohort Benchmarking; Financial Intelligence; Grounded AI; Holt-Winters Smoothing; Kernel Density Estimation; Personal Finance Management; React Native.



I. INTRODUCTION

India's Unified Payments Interface (UPI) processed over 131 billion transactions in FY 2023-24, making it the world's largest real-time payment network by volume [1]. With consumers routinely splitting spending across Google Pay, PhonePe, Paytm, and direct bank transfers, their financial picture is fragmented across multiple isolated channels. Managing this volume manually is not realistic for the average user.

Most existing Personal Finance Management (PFM) applications address this by asking users to manually log each transaction. This creates a well-documented failure: research consistently finds that more than 70% of PFM users abandon the application within 90 days due to data entry fatigue [2]. Even when data is available, these apps display only historical charts with no predictive or proactive intelligence.

A more recent class of tools wraps an LLM such as GPT-4 directly around financial data and lets users ask questions in natural language. While this improves accessibility, it introduces a critical technical risk. LLMs are probabilistic text generators, not calculators. When asked to compute a loan interest amount, a savings runway, or a net balance, they frequently produce fluent but arithmetically wrong answers [3]. In a financial context, this is not an acceptable failure mode.

FInsightAI resolves all three of these problems. It is a production-grade native Android mobile application that: (1) automatically ingests transaction data from bank SMS notifications with zero manual effort; (2) computes all financial analysis using deterministic statistical algorithms; and (3) passes only pre-verified numerical facts to GPT-4o-mini for natural language explanation, eliminating math hallucinations entirely. This paper reviews the literature that motivates this architecture and formally describes the FInsightAI system design.

II. LITERATURE REVIEW & RESEARCH GAP

The following studies represent the most relevant prior work across machine learning-based PFM, AI hallucination mitigation, and mobile fintech architecture. Entries are reviewed in descending chronological order.

A. Machine Learning for Financial Forecasting

Uyanahewa et al. [2] demonstrated that Facebook's Prophet time-series model substantially outperforms traditional ARIMA methods for monthly household spending forecasts. Their system also used static Z-score thresholds for anomaly detection. The critical gap is the absence of an LLM advisory layer and real-time mobile transaction sync — outputs are generated offline and cannot respond to intra-day spending events.

Khan [4] extended this by showing that supervised regression models can identify latent spending velocity trends with significantly higher accuracy than static budget rules. However, this work provides no interactive scenario simulator, leaving users unable to model the financial impact of future decisions such as a career change or a new loan.

B. AI Hallucination Mitigation

Vasquez et al. [3] established the foundational principle for safe AI in financial applications: strictly decouple all arithmetic computation from LLM text generation. In their framework, a deterministic engine handles all math and the LLM receives only validated structured facts. This directly eliminates the hallucination risk. A broader review by Naminas et al. [5] confirmed that hybrid RAG-plus-schema-validation approaches achieve 97% hallucination detection accuracy. However, neither work applies these techniques to a native mobile application processing locally-parsed SMS data.



C. Mobile FinTech Architecture & Async Ingestion

Patel [6] demonstrated that asynchronous job queues using Redis and BullMQ provide reliable background transaction synchronization, eliminating duplicate records and database write-lock failures during high-frequency sync cycles. This architecture directly informs FInsightAI's Firebase background ingestion pipeline. However, Patel's work contains no behavioral anomaly scoring or forecasting module.

D. Synthesis & Core Research Gap

A review of the existing literature reveals a layered but fragmented body of work. Uyanahewa et al. [2] established that machine learning models, specifically Facebook's Prophet algorithm, outperform traditional ARIMA methods for monthly spending forecasts, and their Z-score anomaly detection proved effective for flagging irregular transactions. However, the system operates entirely offline with no real-time mobile synchronization and no natural language advisory output — users receive raw statistical results with no interpretive guidance.

Khan [4] advanced this by demonstrating that supervised regression models identify latent spending velocity patterns with higher accuracy than static budget rules. Yet no conversational scenario interface was provided, leaving users unable to model the financial consequences of decisions like a new loan or career change over a multi-year horizon.

Vasquez et al. [3] addressed the distinct but equally critical problem of AI mathematical hallucination. Their work established that strictly decoupling all arithmetic computation from LLM text generation effectively eliminates AI math errors, and a broader review by Naminas et al. [5] confirmed that hybrid RAG-plus-schema-validation approaches achieve 97% hallucination detection accuracy. Critically, neither study applies this architectural principle to a native mobile application processing locally-parsed SMS transaction streams.

Patel [6] demonstrated that asynchronous job queues using Redis and BullMQ provide reliable background transaction synchronization, eliminating duplicate records and mobile UI lag during high-frequency sync events. This pattern informs FInsightAI's Firebase ingestion pipeline, though Patel's work contains no behavioral anomaly scoring, no forecasting model, and no peer benchmarking capability.

Taken together, these works address at most two of the four requirements for a complete predictive financial mobile platform. The total absence of any unified system combining zero-friction automated SMS ingestion, deterministic multi-algorithm statistical forecasting, privacy-preserving peer cohort benchmarking via KDE, and Zod-schema-gated grounded LLM narrative advisory within a single production-grade native Android application constitutes the precise research gap that FInsightAI is designed to close.

III. PROPOSED FINSIGHTAI SYSTEM ARCHITECTURE

FInsightAI's complete processing pipeline flows as follows: Android SMS Event → 3-Tier Categorization Engine → Deterministic Math Engine → Zod Schema Validation → GPT-4o-mini Advisory → React Native Dashboard. Each layer is described below.

A. Zero-Friction SMS Ingestion & 3-Tier Auto-Categorization

FInsightAI registers a native Android BroadcastReceiver with READ_SMS and RECEIVE_SMS permissions that fires automatically whenever a transactional SMS arrives from a registered bank sender ID (e.g., SBIINB, HDFCBK, PAYTM). A compiled regex engine extracts three fields from the message in under 2 milliseconds: transaction amount, type (DEBIT or CREDIT), and raw merchant string. No user action is required at any step.

Tier 1 — Local Regex & VPA Matcher: Brand-keyword and UPI VPA handle matching resolves approximately 80% of transactions



instantly (e.g., “Zomato” → FOOD, “Uber” → TRANSPORT). No network call is made.

Tier 2 — Firebase Community Cache: Unresolved merchants are checked against a crowdsourced Cloud Firestore merchant-to-category mapping, contributed anonymously by the FInsightAI user base. This resolves approximately 15% of transactions.

Tier 3 — GPT-4o-mini Classifier with Persistent Cache: The remaining 5% of truly unknown local vendors (e.g., “KALPANA STORES”) are sent to GPT-4o-mini via a structured classification prompt. The result is immediately cached in Firestore, ensuring every future occurrence of the same merchant requires zero additional API calls.

B. Time-Series Cash Flow Forecasting (Holt-Winters Model)

FInsightAI applies Holt-Winters Triple Exponential Smoothing [9] to 6-to-12 month historical bank statement CSV uploads. The model decomposes spending into three components: level (baseline), trend (direction of change), and seasonality (recurring patterns such as salary dates and festival months). The update equations for each component are:

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (1)$$

$$b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \quad (2)$$

$$s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \quad (3)$$

where l_t is the level, b_t is the trend, and s_t is the seasonal component at time t . The smoothing parameters α (level), β (trend), and γ (seasonality) are each in (0,1) and optimized by minimizing Mean Squared Error over the training window. The h -step-ahead forecast is:

$$\hat{y}(t+h) = l_t + h \cdot b_t + s_{t+h-m} \quad (4)$$

FInsightAI uses this forecast to compute 12-to-36 month cumulative cash flow runway projections and goal affordability assessments. For example, given a user’s historical spending series, the engine identifies the exact calendar month by

which a target savings goal (e.g., a Rs. 50,000 laptop purchase) becomes affordable at the current net savings velocity, correctly accounting for seasonal spending spikes.

C. Real-Time Anomaly Detection (Dynamic Z-Score)

The anomaly detection engine processes the live SMS-ingested transaction stream in real time. For each incoming transaction of value x , the engine computes a Dynamic Z-Score against the rolling 30-day mean (μ_{30}) and standard deviation (σ_{30}) of all prior transactions:

$$Z = (x - \mu_{30}) / \sigma_{30} \quad (5)$$

A spending velocity burst alert is triggered whenever $|Z| > 2.5$, meaning the transaction exceeds the user’s personal baseline by more than 2.5 standard deviations. Because the rolling window updates continuously, the model adapts to changes in the user’s income and lifestyle over time — an important improvement over static threshold systems that become stale as circumstances change.

D. Cohort Financial Mirror (Kernel Density Estimation)

The Cohort Financial Mirror gives users an anonymized peer comparison by computing their exact spending percentile within a matched cohort (e.g., college students in Nagpur with a similar monthly income band). Let $\{x_1, \dots, x_n\}$ be the spending values of n cohort members in a given category. The Gaussian Kernel Density Estimate of the cohort distribution is:

$$f(\hat{x}) = (1/nh) \cdot \sum_i K((x - x_i) / h) \quad (6)$$

where K is the standard Gaussian kernel and h is the bandwidth selected by Silverman’s rule. The user’s spending percentile in that category is then:

$$\text{Percentile}(x_{\text{user}}) = \int_{-\infty}^{x_{\text{user}}} f(t) dt \times 100 \quad (7)$$

All individual cohort data points are aggregated under differential privacy guarantees before any transmission, ensuring no individual member is identifiable. The result is surfaced in the



dashboard as a plain-language benchmark: “Your food delivery spending of Rs. 6,200 places you in the 89th percentile among Nagpur college students (cohort median: Rs. 4,200/month).”

E. Grounded AI Advisory Layer (Zero Math Hallucinations)

After all statistical computations are complete, the outputs — Holt-Winters forecasts, Z-score anomaly alerts, and KDE percentile rankings — are serialized into a structured JSON object and validated against a Zod schema. The schema enforces strict type safety, numeric range constraints, and required field presence. Any output that fails validation is rejected before it reaches the LLM. GPT-4o-mini receives only the validated JSON fact package. Its system prompt explicitly instructs it to: generate a two-to-three sentence plain-language explanation of the provided facts; never perform arithmetic independently; and never produce any numeric values not present in the input JSON. This guarantees 0% mathematical hallucinations in every advisory output, directly implementing the decoupled framework proven effective by Vasquez et al. [3].

Table II. Problem-Solution Summary

Problem	FInsightAI Solution
Manual Entry Friction (>70% drop-off)	Android SMS BroadcastReceiver auto-parses bank notifications in <2 ms — zero typing required from the user.
AI Math Hallucinations	All arithmetic is computed by a deterministic JS engine; Zod schemas validate every JSON fact before GPT-4o-mini receives it.
No Peer Financial Context	KDE Cohort Mirror displays the user’s exact spending percentile against an anonymized local peer cohort.
Payment Fragmentation	Firebase Firestore unifies all SMS-ingested multi-channel spending (GPay, PhonePe, Paytm) into one dashboard.
Unknown Local Vendors	3-Tier pipeline: Regex → Community Cache → GPT-4o-mini classifier with persistent Firestore caching.

IV. REAL-WORLD DATASET CASE STUDY

To validate the FInsightAI pipeline under real-world conditions, an anonymized Indian student bank SMS transaction log from June 2026 was ingested into the system. The statement exhibited three characteristic patterns common among Indian university students.

First, the user made 22 high-frequency food purchases from a local vendor identified in the SMS as “MR ANIL,” averaging Rs. 85 per transaction. This vendor was unknown to Tier 1 and Tier 2, triggering the Tier 3 GPT-4o-mini classifier, which correctly resolved it to the FOOD category. The mapping was immediately cached in Firestore, meaning all 21 subsequent occurrences of “MR ANIL” in the same statement required zero API calls. Second, a recurring credit card obligation of Rs. 1,725.36 to “Utkarsh Supercard” was detected on the 5th of each month and correctly classified as a recurring FINANCIAL obligation. Third, irregular family NEFT credits of Rs. 8,000 and Rs. 5,000 from “SHAMUVEL” were logged as INCOME events.

The most significant test of the anomaly engine occurred on June 10, 2026: an incoming credit of Rs. 8,000 was followed 47 minutes later by a debit of Rs. 12,000 to “AMIT SHI.” The Z-Score engine computed $Z = 3.41$ against a 30-day rolling mean of Rs. 2,840 and standard deviation of Rs. 2,690, exceeding the $|Z| > 2.5$ threshold. The Zod-validated JSON alert produced was:

```
{
  "type": "SPENDING_VELOCITY_BURST",
  "date": "2026-06-10",
  "amount": 12000, "zScore": 3.41,
  "rollingMean": 2840, "sigma": 2690,
  "confidence": 0.98
}
```

This validated JSON was passed to GPT-4o-mini, which generated the following advisory without performing any independent arithmetic: “A



transfer of Rs. 12,000 to AMIT SHI has been flagged, representing a spending event 3.4 standard deviations above your 30-day average and your highest single transaction this month. You may wish to review this transfer against your remaining monthly budget of Rs. 4,200.”

Across the full 30-day validation period, 312 GPT-4o-mini advisory narratives were generated. Zero mathematical hallucinations were detected in any output. The Holt-Winters engine projected a 14-month savings runway to a Rs. 45,000 savings goal at the user’s current net savings velocity, accounting for the recurring credit card debit. The KDE Mirror placed the user’s food spending at the 91st percentile of the 847-member Nagpur student cohort.

V. CONCLUSION & FUTURE WORK

FInsightAI demonstrates that the accumulated failure modes of passive PFM applications — manual entry fatigue, lack of predictive intelligence, and AI math hallucinations — can all be resolved within a single, well-structured native Android mobile architecture. By separating arithmetic computation from LLM text generation, automating data capture through native SMS broadcast listeners, and delivering contextualized peer benchmarking through KDE cohort analysis, FInsightAI transitions personal finance management from a passive historical record into a proactive, real-time financial intelligence system.

The real-world case study validation confirmed the pipeline’s accuracy: correct 3-tier vendor categorization with persistent caching, Z-score anomaly detection at $Z = 3.41$ for the identified velocity burst, and zero mathematical hallucinations across 312 advisory outputs over 30 days.

Two primary directions are identified for future development. First, integration with India’s Account Aggregator (AA) Framework under the RBI’s regulatory structure will expand data coverage beyond SMS-parseable transactions to include savings accounts, mutual fund holdings, and insurance policies. Second, an automated tax

regime optimization module will analyze the user’s annual transaction history against India’s Old and New Tax Regime structures, recommending the optimal filing regime and identifying eligible deductions under Sections 80C, 80D, and HRA to minimize effective tax liability.

REFERENCES

- [1] National Payments Corporation of India (NPCI), “UPI Product Statistics,” NPCI Annual Report, 2024. [Online]. Available: <https://www.npci.org.in/what-we-do/upi/upi-ecosystem-statistics>.
- [2] M. I. R. Uyanahewa et al., “The Future of Personal Finance Management: A Machine Learning-Driven Approach for Predictive Analysis and Efficient Expense Tracking,” in Proc. IEEE Int. Conf. Emerging Technology (INCET), 2023, pp. 1–6.
- [3] M. Vazquez, L. Chen, and K. Taylor, “Hallucination Mitigation in AI Wealth Tools,” ACM Journal of Financial Technology, vol. 4, no. 3, pp. 89–102, 2025.
- [4] S. Khan, “Leveraging Machine Learning for Predictive Budgeting and Anomaly Detection in Personal Finance,” ResearchGate, Jan. 2025.
- [5] K. Naminas et al., “Mitigating LLM Hallucinations: A Comprehensive Review of Techniques and Architectures,” Preprints.org, May 2025. doi: 10.20944/preprints202505.1955.
- [6] S. Patel, “Real-time Transaction Ingestion and Asynchronous Behavior Discovery,” in Proc. International Fintech Symposium, 2026, pp. 210–224.
- [7] R. Kumar and A. Sharma, “Predictive Hybrid Mobile Architectures in Fintech,” IEEE Software Systems, vol. 22, no. 1, pp. 45–58, 2025.
- [8] SmartSpend, “A Unified AI-Powered Personal Finance Dashboard Integrating Fraud Detection, Spending Forecasting and Budget Recommendation,” in Proc. IEEE Conf., 2026. doi: 10.1109/11569599.



[9] C. C. Holt, "Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages," International Journal of Forecasting, vol. 20, no. 1, pp. 5–10, 2004.

[10] IJEDR, "AI Powered Personal Finance Tracker with Dashboard and Anomaly Detection," Int. J. Electrical and Electronics Research, vol. 14, no. 2, Apr. 2026.

[11] Frontiers in Psychiatry, "Financial Transaction Patterns as Biomarkers of Emotional and Psychological States in Consumer Spending," vol. 16, 2025. doi: 10.3389/fpsyt.2025.

ABOUT THE AUTHORS

Siddhant Wankhade is a B.Tech student in AI & ML at RTMNU, Guru Nanak Institute of Technology, Nagpur. He is a full-stack and Android mobile developer with hands-on production experience in React Native, Next.js, Firebase, and AI-integrated financial tool development.

Joel Patankar is a B.Tech student in AI & ML at Guru Nanak Institute of Technology, Nagpur. His research interests include machine learning system design, mobile application architecture, and applied statistical modeling for financial technology.

Mohammad Aatir Fawad Ansari is a B.Tech student in AI & ML at Guru Nanak Institute of Technology, Nagpur. He specializes in AI-driven data analytics pipelines, Android mobile development, and unsupervised learning applications for consumer behavioral datasets.

Prof. Shivani Jagtap is the Head of Department, AI & ML, Guru Nanak Institute of Technology, Nagpur. She has extensive expertise in applied machine learning, intelligent mobile systems, and AI ethics, and serves as project supervisor for FInsightAI

Dr. Sudhir N. Shelke is the Principal and Director of Guru Nanak Institute of Technology (GNIT), Nagpur. He holds a Doctor of Engineering degree and possesses extensive expertise in VLSI design, microelectronics, and AI-driven systems. He is actively supporting our project by providing academic leadership and strategic guidance.