

ProcureAi: An AI-Driven Procurement Spend Intelligence Platform For Automated Anomaly Detection, Supplier Optimisation And Predictive Expenditure Forecasting

Bharrath K

Dept of Computer Science & Engineering
Rajalakshmi Engineering College, Chennai, India

Abstract- Enterprises routinely lose between three and eight percent of addressable procurement spend to fragmentation, duplicate payments, maverick buying, contract price leakage, and unmanaged tail spend. This paper presents ProcureAI, a full-stack, AI-driven spend-intelligence platform that ingests multi-source procurement data, classifies it automatically using a hybrid NLP classifier combining Multinomial Naive Bayes with TF-IDF cosine k-NN re-ranking, detects leakage through a six-detector unsupervised anomaly ensemble, segments the supplier base with K-Means++ clustering augmented by Herfindahl-Hirschman Index concentration analysis, forecasts future expenditure via a four-model inverse-MAPE-weighted time-series ensemble with residual-quantile prediction intervals, and synthesises all signals into a ranked, costed, auditable cost-reduction portfolio. Applied to a 36-month synthetic enterprise dataset of ~8,000 transactions and \$85 million in total spend, the platform achieves ~86% NLP classification accuracy (macro-F1 ~87%), detects ~4,000 anomaly events, achieves a forecasting ensemble MAPE of ~18%, and identifies \$7.3 million (8.4%) in annual run-rate savings. All algorithms are implemented from first principles in TypeScript, ensuring full auditability without opaque model binaries or external ML services.

Keywords- Procurement Analytics, NLP Spend Classification, Anomaly Detection, Supplier Segmentation, K-Means++, Isolation Forest, Holt-Winters, Time-Series Forecasting, Cost Optimisation, Spend Intelligence

I. INTRODUCTION

Modern enterprises operate across dozens of business units, geographic regions, and heterogeneous ERP platforms—SAP, Oracle, Ariba, Coupa, Infor, Microsoft Dynamics, and bespoke legacy systems—each generating procurement transactions in isolated silos. The resulting data fragmentation means that category managers, finance directors, and CFOs routinely make sourcing decisions without a unified, coherent view of what the organisation is

buying, from whom, at what price, and under what contractual conditions.

Industry benchmarking studies consistently report that unmanaged procurement leakage consumes three to eight percent of total addressable spend. Five canonical pathologies account for the bulk of this leakage: (i) vendor fragmentation; (ii) duplicate invoice payments; (iii) maverick purchasing; (iv) contract price leakage; and (v) tail-spend rationalisation failures.

ProcureAI bridges this gap with a five-module pipeline: (1) multi-ERP data integration and NLP spend categorisation; (2) six-detector anomaly detection; (3) vendor consolidation and cost optimisation; (4) predictive procurement analytics with ensemble forecasting; and (5) ranked recommendation generation with phased savings realisation schedules.

II. RELATED WORK

Early procurement classification systems employed rule-based UNSPSC taxonomies. Probabilistic text classification with Naive Bayes and SVMs followed, with Faure et al. reporting 78–82% accuracy. Transformer-based models (BERT, GPT variants) achieve higher accuracy but introduce model opacity and latency incompatible with real-time audit requirements. The hybrid NB+TF-IDF approach used here deliberately trades 3–6% accuracy for full transparency and sub-millisecond inference.

Anomaly detection is dominated by Isolation Forest (Liu et al., 2008) for multivariate numeric anomalies and MAD-based z-scores (Iglewicz & Hoaglin, 1993) for univariate price outliers. Supplier segmentation follows the K-Means++ framework (Arthur & Vassilvitskii, 2007) augmented by the Herfindahl-Hirschman Index. Forecasting ensembles draw on the M4 Competition findings (Makridakis et al., 2020).

III. SYSTEM ARCHITECTURE

ProcureAI is structured as a four-layer pipeline within a Next.js 16 (App Router) full-stack application backed by PostgreSQL with 13 relational tables and 9 covering indexes.

A. Data Layer: Performs multi-ERP ingestion with five normalisation steps: currency conversion (USD), UoM harmonisation, supplier name deduplication (Soundex + token-set ratio), transaction deduplication (composite hash key), and data quality profiling.

B. AI Engine: Five ML pipelines execute as serverless API route handlers: NLP Classifier, Anomaly Ensemble, Supplier Segmentation, Expenditure Forecaster, and Recommendation Generator.

C. Analytics Layer: Exposes a multidimensional spend cube across taxonomy category, business unit, geographic region, ERP platform, supplier tier, and calendar period dimensions.

D. Decision Layer: React 19 frontend renders eight analytical workspaces using 18 bespoke SVG/Canvas chart primitives built without any third-party charting library.

Table I: Technology Stack

Layer	Technology
Core Framework	Next.js 16, React 19, TypeScript 5.9
Database	PostgreSQL via Drizzle ORM (13 tables, 9 indexes)
ML Algorithms	Pure-TypeScript: NB, TF-IDF, Isolation Forest, K-Means++, Holt-Winters, OLS, Ridge, Jaro-Winkler, LSH
Container	Docker multi-stage build (~180 MB)
CI/CD	GitHub Actions: lint → tsc → build → Trivy CVE scan → deploy

IV. METHODOLOGY

A. Module 1 – NLP Spend Classification: Two-stage hybrid classifier. Stage 1: Laplace-smoothed Multinomial Naive Bayes. Stage 2: TF-IDF cosine similarity re-ranking. Final score: $0.45 \cdot P_{NB} + 0.55 \cdot \text{cosine}(\text{TF-IDF, centroid})$. Shannon entropy H flags uncertain predictions for human review.

B. Module 2 – Six-Detector Anomaly Ensemble: Detector 1: Isolation Forest (120 trees, $s > 0.65$). Detector 2: Robust MAD z-score per (category $\times$ UoM) pair, $|M_i| > 3.5$. Detector 3: MinHash LSH + Jaro-Winkler for fuzzy duplicate detection. Detector 4: Logistic regression maverick classifier. Detector 5: Deterministic contract leakage comparison. Detector 6: Seasonal volume spike (2.5σ rule).

C. Module 3 – Supplier Segmentation: Composite supplier score $S = 0.30 \cdot P_{\text{perf}} + 0.25 \cdot P_{\text{price}} + 0.20 \cdot R_{\text{rel}} + 0.15 \cdot (1 - R_{\text{risk}}) + 0.10 \cdot \text{ESG}$. K-Means++ ($k=4$) segments into 4 strategic tiers. HHI concentration analysis. Ultimate-parent roll-up reveals hidden concentration.

D. Module 4 – Forecasting Ensemble: Four models: Holt-Winters triple exponential smoothing, Decomposition + OLS seasonal dummies, Damped-Drift seasonal random walk ($\phi=0.92$), Ridge-regularised neural forecaster. Blended by inverse-MAPE weights.

E. Module 5 – Recommendation Engine: Savings velocity $V = (\text{savings} \times \text{confidence}) / (1 + \text{effort})$. Ranked descending by V . Phased realisation: 15% Q1 / 35% Q2 / 30% Q3 / 20% Q4.

V. EXPERIMENTAL RESULTS

All experiments conducted on a deterministic 36-month synthetic enterprise dataset (seed=42): ~8,000 transactions, \$85M total spend, 16 BUs, 6 regions, 7 ERPs, 20 taxonomy nodes, 160 suppliers, 40+ contracts.

Table II: NLP Classifier Performance

Metric	Hybrid NB+TF-IDF	NB Only	TF-IDF Only
Hold-out Accuracy	~86%	~79%	~83%
Macro-F1 Score	~87%	~80%	~84%
Top-3 Accuracy	~97%	~93%	~95%
Avg. Inference Time	<1 ms	<0.5 ms	<1 ms

Table III: Anomaly Detection Results

Detector	Flags	Est. Financial Impact
Isolation Forest	~650 txns	~\$1.2M
Robust MAD z-score	~820 price outliers	~\$0.9M
Fuzzy Duplicate LSH	~90 duplicate clusters	~\$0.6M
Maverick Classifier	~26% of spend	~\$22M at risk
Contract Leakage	~340 line items	~\$4.1M
Seasonal Volume Spike	~180 anomalies	~\$0.4M
Total (Ensemble)	~4,000 events	~\$7.3M actionable

Table IV: Forecast Model Performance (Rolling-Origin Back-Test)

Model	MAPE	RMSE	Dir. Accuracy
Holt-Winters	~21%	\$142K	72%
Decomposition + OLS	~22%	\$151K	70%
Damped-Drift RW	~25%	\$168K	68%
Ridge Neural	~20%	\$135K	74%
Ensemble (inv-MAPE)	~18%	\$121K	76%

Table V: Annual Savings Portfolio

Recommendation Family	Est. Annual Savings	% of Spend
Contract Leakage Recovery	\$4.1M	4.8%
Vendor Consolidation	\$1.4M	1.6%
Maverick Spend Control	\$0.7M	0.8%
Duplicate Payment Recovery	\$0.6M	0.7%
Price Benchmark Arbitrage	\$0.3M	0.4%
Tail-Spend Rationalisation	\$0.2M	0.2%
Grand Total	\$7.3M	8.4%

VI. DISCUSSION

The 86% accuracy without deep learning is noteworthy. Domain-specific procurement vocabulary provides strong discriminative signal captured effectively by TF-IDF cosine similarity. The six-detector ensemble heterogeneity is crucial: single-detector coverage would miss entire pathology classes. The TypeScript-native ML implementation yields a ~180 MB Docker image vs. ~2.5 GB for a comparable Python ML stack, with sub-100 ms API responses from cache. Key limitations include synthetic dataset origin, English-only NLP, and hand-crafted logistic regression weights for the maverick classifier.

VII. CONCLUSION

ProcureAI comprehensively addresses the five canonical spend-leakage pathologies within a single, auditable, full-stack system. Applied to a 36-month synthetic enterprise estate, the platform identified \$7.3 million (8.4%) in annual run-rate savings through a hybrid NLP classifier (~86% accuracy), six-detector anomaly ensemble, K-Means++ supplier segmentation, and a four-model forecasting ensemble (~18% MAPE). The TypeScript-native ML kernel ensures full auditability and operational simplicity without external ML service dependencies.

VIII. ACKNOWLEDGMENT

The author thanks the faculty of the Department of Computer Science & Engineering, Rajalakshmi Engineering College, for their guidance throughout the capstone project, and the open-source community whose libraries enabled the rapid development of a production-grade system.

REFERENCES

- [1] CAPS Research, "Benchmarking: Accounts Payable and Procurement," Arizona State University, 2022.
- [2] R. Crandall and M. Crandall, "New Methods of Competing in the Global Marketplace," CRC Press, 2009.
- [3] UNSPSC Secretariat, "United Nations Standard Products and Services Code: Reference Document," UNSPSC, 2023.
- [4] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.
- [5] M. Faure, T. Comes, and D. Hiete, "Data-driven spend classification using machine learning," *Proc. ICIS*, 2019.
- [6] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers," *Proc. NAACL-HLT*, 2019.
- [7] M. M. Breunig et al., "LOF: Identifying Density-Based Local Outliers," *Proc. ACM SIGMOD*, 2000.
- [8] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," *Proc. ICDM*, 2008.
- [9] B. Iglewicz and D. C. Hoaglin, "How to Detect and Handle Outliers," ASQC Quality Press, 1993.
- [10] [A. Gionis, P. Indyk, and R. Motwani, "Similarity Search in High Dimensions via Hashing," *Proc. VLDB*, 1999.
- [11] P. Kraljic, "Purchasing must become supply management," *Harvard Business Review*, vol. 61, no. 5, 1983.
- [12] [D. Arthur and S. Vassilvitskii, "k-means++: The Advantages of Careful Seeding," *Proc. SODA*, 2007.
- [13] O. C. Herfindahl, "Concentration in the U.S. Steel Industry," Ph.D. dissertation, Columbia Univ., 1950.
- [14] C. C. Holt, "Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages," *Int. J. Forecasting*, 2004.
- [15] S. Makridakis et al., "The M4 Competition: 100,000 time series and 61 forecasting methods," *Int. J. Forecasting*, 2020.
- [16] H. Katzorke et al., "Procurement forecasting with exponential smoothing," *J. Purchasing Supply Mgmt.*, 2016.
- [17] A. Nair, J. Jayaram, and A. Das, "Strategic purchasing participation," *Int. J. Production Research*, 2015.
- [18] B. N. Lim et al., "Temporal Fusion Transformers," *Int. J. Forecasting*, vol. 37, no. 4, 2021.
- [19] R. J. Hyndman and G. Athanasopoulos, "Forecasting: Principles and Practice," 3rd ed., OTexts, 2021.
- [20] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning," 2nd ed., Springer, 2009.