IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

Machine Learning-Based Procurement Price Benchmarking Through Historical Ranking and Market Evidence

Keshava A[1], Amrutha N V[2], Anu Patel M V[3], Deepthi M[4], Shashank S[5]

Assistant Professor, ECE, Bangalore Institute of Technology, Bangalore, India[1]

Undergraduate Student, ECE, Bangalore Institute of Technology, Bangalore, India[2]

Undergraduate Student, ECE, Bangalore Institute of Technology, Bangalore, India[3]

Undergraduate Student, ECE, Bangalore Institute of Technology, Bangalore, India[4]

Undergraduate Student, ECE, Bangalore Institute of Technology, Bangalore, India[5]

Abstract: Historical procurement records can support price benchmarking, but selecting a suitable reference is challenging when similar descriptions represent different brands, models, specifications, commercial units, quantities, or procurement contexts. This paper examines the historical candidate-reranking component developed for procurement price decision support in CivicEngage. For each query, an upstream retrieval stage supplies ten historical candidates. The proposed reranker combines semantic similarity, structured-attribute matching, retrieval signals, and historical price-consensus features using a LightGBM binary classifier with a nested-cross-validation-based decision gate. It either retains the original Rank-1 candidate or replaces it with a candidate from the supplied set; the returned benchmark is the selected candidate's original unit price, not a newly generated or adjusted price. On the documented benchmark of 10,560 queries and 105,600 candidate rows, the gated reranker achieved 38.99% Accuracy@10, compared with 37.57% for the original Rank-1 baseline, an improvement of 1.42 percentage points. On a clean subset of 9,417 queries whose target descriptions did not contain explicit price strings, it achieved 39.70%, compared with 38.36% for the baseline. These results show a measurable but limited improvement for the evaluated candidate pool. They do not establish general current-market price accuracy, and the result remains constrained by whether a suitable historical candidate is retrieved.

Keywords: Procurement Price Benchmarking; Historical Procurement Data; Candidate Reranking; LightGBM; Binary Classification; Semantic Similarity; Price Consensus; Decision Support

I. INTRODUCTION

A. Background and Motivation

Procurement price benchmarking helps organizations assess whether a proposed unit price is consistent with relevant historical transactions. For non-governmental organizations (NGOs), this evidence can support budget preparation and tender decisions. However, selecting a useful historical reference is not straightforward. Records with similar textual descriptions may refer to different brands, models, technical specifications, quantities, units of measure, or purchasing contexts. Their prices may therefore differ substantially even when the wording appears similar.

The Procurement Benchmark component in CivicEngage addresses this problem by selecting a reference from historical procurement candidates. An upstream retrieval process supplies ten candidates for each query. The candidate-reranking component evaluates those records using multiple types of evidence and selects one candidate whose original unit price is returned as the benchmark reference. This bounded selection approach keeps the output traceable to a recorded transaction and avoids presenting an adjusted or generated value as though it were an observed purchase price.

The quality of the result depends on both retrieval and selection. A reranker can improve the order of retrieved records, but it cannot select an appropriate reference if no suitable record is present in the candidate set. This motivates an approach that combines semantic and structured evidence while measuring performance against the original Rank-1 choice.

B. Problem Statement

Given a procurement query and ten historical candidate records supplied by the retrieval stage, the objective is to choose a candidate whose original unit price is within a defined tolerance of the observed unit price used as evaluation ground

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

truth. The implemented system uses a ±10% tolerance as its primary success criterion. For a non-zero ground-truth price y_i and selected candidate price p_i, a query is counted as successful when $|p_i - y_i| / |y_i| \leq 0.10$. Accuracy@10 is the percentage of evaluation queries satisfying this condition.

The decision is not an unconstrained regression that independently generates a price. It is a candidate-selection problem: the system either keeps the original Rank-1 result or replaces it with another supplied candidate. Evaluation-only values, including the ground-truth price and error/outcome labels, must not be used as selection-time features. Explicit price and currency strings in target descriptions are removed before text-derived feature extraction to address this known source of leakage. This control does not, by itself, prove that every possible leakage pathway has been eliminated.

C. Research Objectives

To establish the original Rank-1 candidate-selection result as a baseline for historical procurement price benchmarking.
To combine semantic similarity, structured-attribute matching, retrieval information, and historical price-consensus signals when assessing candidate suitability.
To implement a LightGBM binary classifier with a nested-cross-validation-based decision gate that determines whether to retain the baseline candidate or replace it.
To evaluate the reranker using the ±10% Accuracy@10 metric and supporting analyses, including a clean-subset evaluation.
To distinguish candidate-selection performance from current-market price prediction and recognize the limitations imposed by the retrieved candidate pool.

D. Research Contributions

The contribution reported here is the historical procurement candidate-reranking component of CivicEngage. First, it formulates benchmark selection as a bounded decision over ten retrieved historical candidates and returns the selected record's original unit price. Second, it combines semantic, structured-attribute, retrieval, and historical price-consensus features in a LightGBM binary classifier. Third, a nested-cross-validation-based decision gate is used to decide whether a candidate replacement should be made rather than automatically replacing the baseline choice for every query.
On the documented 10,560-query benchmark, the gated reranker achieved 38.99% Accuracy@10 compared with 37.57% for the original Rank-1 baseline, a gain of 1.42 percentage points. The clean-subset result was 39.70% compared with 38.36% for the baseline on 9,417 queries. These figures demonstrate a limited improvement on the evaluated data, not near-perfect accuracy or generalization to all procurement settings. The contribution is an applied integration and evaluation of established techniques; it is not a claim that the underlying algorithms themselves are newly invented.

II. LITERATURE REVIEW

A. Existing Procurement Price Estimation Approaches

Procurement pricing has been studied using pricing models and machine-learning methods. Herrmann and Cobo Martín discuss pricing models in business-to-business and business-to-government tendering, including the gap between proposed models and their practical adoption [1]. García Rodríguez et al. analyse public procurement announcements and develop a machine-learning estimator for tender award prices using tender-related attributes [2]. These studies establish the relevance of data-driven procurement pricing and show that historical procurement information can be used to model price-related outcomes.

The present work addresses a related but narrower task. Tender-level price estimation predicts an award price from tender characteristics, whereas this component selects one historical item-level price from a fixed candidate set. Its purpose is to identify a sufficiently comparable observed transaction, not to produce an independent price estimate. This distinction is important because a price reference can be misleading when item specifications, units, quantities, or procurement contexts are not comparable.

B. Machine Learning for Procurement Benchmarking

Machine-learning methods can combine multiple signals from historical purchasing records, but the usefulness of a benchmark depends on the quality and relevance of the evidence. Semantic similarity can identify descriptions that use related wording, while structured attributes can expose differences in product characteristics and commercial details. Retrieval signals and historical price-consensus information provide additional context for judging whether a candidate is a plausible reference.

Balkan and Akyuz review artificial-intelligence and machine-learning applications in procurement and purchasing decision support, identifying a broad set of opportunities and research directions [3]. Kuchkorov et al. investigate

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

machine-learning price prediction from laptop specifications, illustrating how product attributes can inform price analysis in a different domain [4]. These works support the general motivation for using structured and descriptive features, but neither should be described as the specific candidate-reranking method implemented in this CivicEngage component.

For reliable evaluation, outcome-derived information must be kept separate from selection-time inputs. In this project, the observed target price and outcome/error labels are evaluation-only, and explicit price strings are removed from target descriptions before text-feature extraction. The clean-subset evaluation examines performance after excluding target descriptions with explicit price strings; it is a targeted leakage check, not proof that every possible form of leakage is absent.

C. Retrieval and Learning-to-Rank Approaches

Retrieval and reranking perform different roles in candidate-based decision support. Retrieval supplies records that may be relevant to the query; reranking assesses those candidates more carefully and decides which should be placed first. Keeping these stages distinct makes the source of an error easier to reason about: a suitable record may be absent from the retrieved set, or it may be present but ranked below a less suitable candidate.

The implemented CivicEngage component uses a LightGBM binary classifier with a nested-cross-validation-based decision gate. The classifier combines semantic similarity, structured-attribute matching, retrieval signals, and historical price-consensus features. The gate determines whether to retain the original Rank-1 candidate or replace it with another supplied record. This design is conservative in scope: the system selects from the retrieved candidates and preserves the selected record's original unit price. It does not apply direct quantity- or time-based adjustments to manufacture a new point prediction.

The reported benchmark contains 10,560 queries and 105,600 candidate rows, with ten candidates per query. The primary gated reranker achieved 38.99% Accuracy@10, compared with 37.57% for the baseline. On a clean subset of 9,417 queries, the corresponding results were 39.70% and 38.36%. These results are specific to the documented candidate pool and evaluation setup; they should not be interpreted as current-market price accuracy.

D. Research Gap

Existing research provides relevant foundations for tender-level price estimation [1], [2], broader machine-learning-based procurement decision support [3], and product-attribute-based price prediction [4]. However, tender-price estimation and general price prediction do not directly resolve the narrower operational question considered here: how to select a suitable historical item-level price from a fixed set of retrieved candidates while preserving traceability to the original transaction. The gap addressed by this work is the applied evaluation of a candidate-reranking decision for historical procurement benchmarking that combines semantic, structured, retrieval, and price-consensus evidence and can decline to replace the baseline candidate. The evaluation also distinguishes the primary ±10% selection metric from the clean-subset leakage check and recognizes that reranking is constrained by retrieval quality. This is an integration-and-evaluation contribution, not a claim that no previous research has used similar machine-learning techniques.

III. DATASET AND PROBLEM FORMULATION

A. Dataset Source and Description

The proposed procurement benchmarking framework identifies comparable historical records to establish evidence - backed reference prices for procurement requests. The historical catalog comprises 209,580 records and 75 columns, representing the available dataset dimensions rather than the number of training queries or final model features. Relevant attributes include item descriptions, specifications, categories, units, prices, and source identifiers, subject to the actual catalog schema. A structured current-market evidence schema is also defined, comprising item name, specification, unit, observed price, currency, source, URL, collection date, location, and notes. These attributes facilitate provenance tracking and comparability assessment; however, schema availability alone does not establish complete data collection or integration into the inference pipeline.

B. Data Preprocessing

Preprocessing aims to improve record consistency while preserving price-sensitive product attributes. Relevant operations include handling missing descriptions, duplicate records, invalid prices, inconsistent units, and incomplete specifications. Text normalization may include case folding, whitespace normalization, punctuation handling, and abbreviation standardization. Model identifiers, dimensions, capacity, material, grade, and packaging information must be retained to prevent false equivalence. Unit normalization is performed only when semantic and quantitative equivalence is established. Original record identifiers and source metadata are preserved to support traceability and

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

reproducibility. Current-market observations retain their source URLs, collection dates, currencies, units, and locations. Only preprocessing operations verified in the implementation are reported as completed.

C. Unit-Price Benchmarking Formulation

Let q denote a procurement query and $C(q) = \{c_1, c_2, \ldots, c_k\}$ the retrieved candidate set. Each candidate contains available descriptive attributes, specifications, unit information, and price evidence. For a source total price P_i associated with a compatible quantity $Q_i > 0$, the unit price is $p_i = P_i / Q_i$. An already validated unit price is used directly. Currency and unit associations are retained throughout processing.

The benchmark is defined as The overall candidate-reranking workflow is shown in Fig. 1.
$\hat{p}(q) = p(c^*)$ where $c^* \in C(q)$ is a selected candidate satisfying semantic compatibility, unit consistency, price validity, and provenance requirements. Textual similarity alone is insufficient for candidate acceptance. If no candidate satisfies the evidence constraints, the system returns an insufficient-evidence outcome rather than an unsupported estimate.

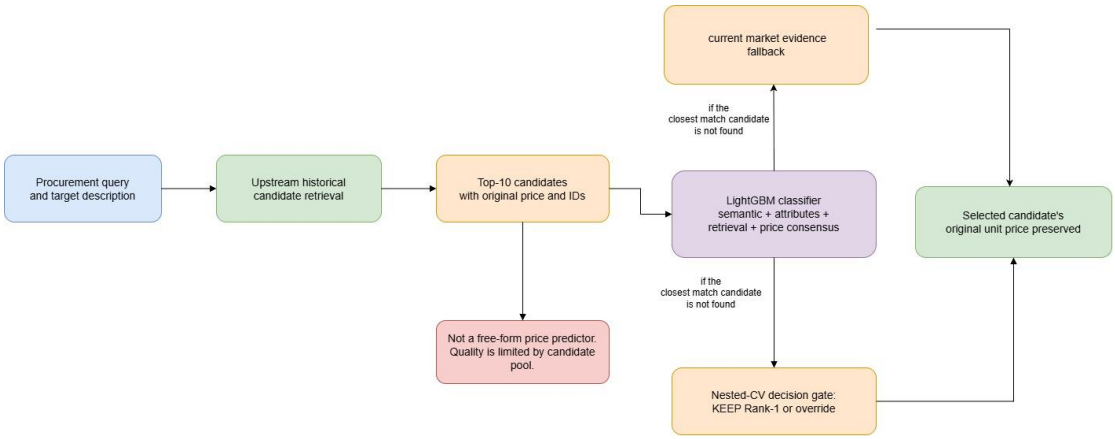


Fig. 1: Procurement Candidate Reranking

D. Data Splitting and Leakage Prevention

Evaluation follows the project's frozen protocol, with partition ratios, query counts, and grouping rules obtained from the corresponding evaluation artifacts. Records originating from the same procurement request, source document, or closely related item descriptions should be grouped where necessary to minimize cross-partition leakage. Data-dependent preprocessing, feature selection, model fitting, and parameter tuning must remain confined to the designated training and validation partitions. Test references and relevance labels are reserved exclusively for evaluation and must not influence retrieval, ranking, or LLM inference. The final manuscript should report the verified split configuration and protocol version.

IV. PROPOSED METHODOLOGY

A. Pipeline Architecture

The proposed framework integrates query normalization, historical candidate retrieval, machine - learning ranking, LLM-based semantic reranking, evidence validation, and benchmark generation.
The pipeline preserves candidate identifiers and source metadata across stages to ensure traceable price selection. Following query validation, the retrieval module identifies potential comparators, and the ranking model orders them by relevance. The leading candidates undergo semantic assessment and subsequent validation of specifications, units, price availability, and provenance. Validated historical evidence supports benchmark generation; an enabled and validated market -evidence pathway may serve as a fallback. If neither source provides adequate support, the system abstains. The relationships among the main application entities are illustrated in Fig. 2.

B. Feature Engineering and Product Representation

Candidate representation combines textual descriptors with available structured attributes. Potential features include normalized name similarity, specification overlap, category agreement, unit compatibility, and numerical product attributes. The final feature set must conform to the implemented model and frozen feature schema. Textual representations facilitate matching across terminological variations, while structured attributes distinguish products with materially different specifications. Consistent feature extraction, feature ordering, and missing-value handling are required across training and inference. Missing attributes are not treated as evidence of equivalence.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

C. Candidate Retrieval and Ranking

Candidate retrieval reduces the historical catalog to a manageable subset using the mechanisms implemented in the project. Retrieved records retain their original descriptions, specifications, identifiers, units, and price fields for downstream processing. For query q and candidate c_i, the frozen ranking model computes $s_i = f\theta(q, c_i)$, where $f\theta$ denotes the trained ranking function and s_i its relevance score. Candidates are ordered by score, and the leading ten records form the Top-10 candidate set supplied to the semantic reranker. Ranking scores represent relative relevance, not necessarily calibrated probabilities of correctness. Final acceptance remains conditional on attribute compatibility and price-evidence validation.

D. LLM-Based Semantic Reranking

The LLM evaluates the Top-10 candidates to identify a semantically compatible record or reject the available evidence. Assessment considers item identity, critical specifications, qualifiers, and unit compatibility, reducing false matches caused by superficial textual similarity. The LLM operates under a constrained output schema, returning a supplied candidate identifier or an explicit rejection outcome. Output validation rejects malformed responses and identifiers outside the candidate set. The selected record's original price and provenance are preserved; the LLM does not independently generate or modify prices. Semantic agreement alone does not establish price validity. Missing or contradictory attributes may require abstention. Compatibility testing between the reranking component and the current frozen Top-10 artifacts remains necessary before claiming end-to-end verification.

E. Benchmark Generation and Uncertainty Handling

The benchmark is derived from the selected candidate's validated price evidence, retaining the associated currency, unit, and source record. Total prices are converted to unit prices only when the corresponding quantity and unit are compatible. Where supported by the implementation, the response may include an expected price range and a reliability indicator. Empirical ranges must be distinguished from statistically calibrated confidence intervals, and heuristic reliability scores must not be interpreted as calibrated probabilities without validation. Insufficient price evidence, incompatible units, contradictory specifications, or weak comparability may trigger rejection or fallback handling. This evidence -constrained approach prioritizes defensible reference prices over unrestricted numerical prediction.

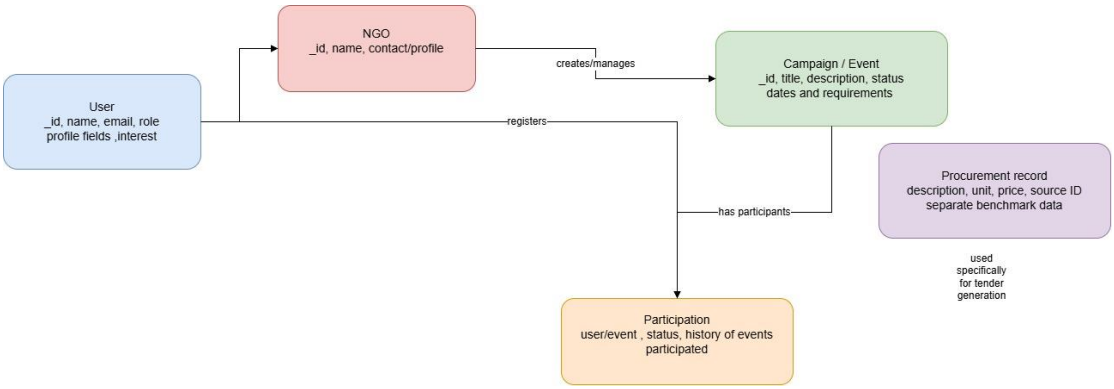


Fig. 2: Conceptual data model

F. End-to-End Integration

The integration layer connects retrieval, ranking, semantic reranking, evidence validation, and response serialization through defined interfaces. It handles empty candidate sets, missing prices, invalid LLM outputs, unit conflicts, and service failures explicitly. Candidate identifiers and source metadata remain linked to the final response, enabling auditability and independent component testing.

V. SYSTEM IMPLEMENTATION

A. Data Processing and Ranking

The implementation uses a historical catalog containing 209,580 records and 75 columns. Data validation and preprocessing prepare model-compatible representations while preserving original identifiers and source values. The actual preprocessing operations, feature schema, training parameters, and partition configuration must be reported from the current codebase and frozen artifacts. During inference, the query is transformed into the representation expected by the retrieval component. Retrieved candidates are scored using the frozen ranking artifact, and the leading candidates are forwarded for semantic reranking. Empty retrieval results, invalid prices, and incompatible units require explicit handling.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

B. LLM and API Integration

The LLM service evaluates ranked candidates using a constrained prompt and structured response format. The response-processing layer validates candidate identifiers and rejects malformed or unsupported fields. Benchmark prices are retrieved from the selected source record rather than generated by the LLM. The ML service exposes /health for service-status checks and /predict for prediction requests. The prediction endpoint is intended to execute the configured pipeline and return a structured response. Its request schema, response fields, error handling, and operational status must be confirmed against the current implementation and test logs. The wider CivicEngage application communicates with the ML service through its configured service URL. Integration verification should cover request serialization, response parsing, service availability, timeout handling, and insufficient-evidence outcomes. Compatibility between the LLM reranker and the frozen Top-10 artifacts must be established before claiming complete pipeline validation.

C. Evidence Management and Output Generation

The output layer returns a benchmark only when the selected record satisfies the required evidence constraints. Depending on the verified API contract, the response may contain the benchmark unit price, expected range, reliability indicator, candidate identifier, and source metadata. The current-market evidence schema supports provenance through item specifications, units, observed prices, currencies, source URLs, collection dates, and locations. Its operational use as a fallback must be verified independently; defining the schema does not establish that the evidence is populated, validated, or integrated. When neither historical records nor an enabled market -evidence pathway provides a defensible comparator, the service returns an insufficient-evidence outcome. Validation failures, fallback events, and service errors should be recorded separately to disting uish successful benchmarking from deliberate abstention. The final implementation claims are restricted to verified functionality. In particular, frozen-artifact compatibility, API behavior, and current-market fallback integration must be confirmed before the complete pipeline is reported as operational.

VI. EXPERIMENTAL SETUP AND EVALUATION

A. Experimental Configuration

Experiments used a frozen historical procurement benchmark containing 10,560 queries and ten retrieved candidates per query, totalling 105,600 candidate rows. Each query was evaluated against the same Top-10 pool to make reranking methods comparable. Candidate-side fields included candidate rank, description, unit price and retrieval score. Available target-side fields included description, quantity, unit of measure, brand, model, commodity code, commodity family, procurement date and location.

The primary learned reranker used LightGBM binary classification with 77 engineered features and sentence-transformers/all-MiniLM-L6-v2 for text representation. The reported configuration used 300 estimators, learning rate 0.05, 31 leaves, minimum child samples 40, row subsampling 0.8, feature subsampling 0.8, L2 regularization 1.0 and random seed 42. The final policy combined candidate scores with a Rank-1-aware gate whose margin was selected through nested cross-validation.

Evaluation used five folds grouped by commodity family (20 groups). This grouping was intended to reduce dependence on random distribution of records from the same commodity family across training and evaluation folds. The fold-specific gate margins were 0.03, 0.03, 0.00, 0.03 and 0.02. The frozen benchmark was not modified.

B. Baseline Methods

The Rank-1 baseline selected the first candidate returned by the upstream retrieval system. Oracle@10 was computed by checking whether any of the ten candidates satisfied the correctness tolerance; it is a theoretical upper bound for selectors restricted to this pool, not an achieved model score.

Compared methods included semantic embedding argmax; a structured attribute scorer using brand, model and commodity-code matches; a semantic–attribute hybrid; a cross-encoder; an LLM-based reranker; a multi-aspect scorer combining semantic, attribute, historical price-consensus and retrieval evidence; a hand-designed Rank-1 KEEP/OVERRIDE gate; LightGBM binary argmax; LightGBM LambdaRank argmax; and the primary LightGBM binary model with nested-CV gate.

C. Evaluation Metrics

The primary metric was Accuracy@±10% (reported in the experiment logs as Accuracy@10): a selected candidate was correct when its unit price was within ±10% of the actual unit price. Additional metrics included Accuracy@5 and Accuracy@20 where reported, mean absolute error (MAE), median absolute percentage error (MdAPE), capped mean absolute percentage error, override rate, recovered cases, damaged cases and net gain. A recovered case is a Rank-1 error

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

corrected by reranking; a damaged case is a correct Rank-1 selection changed to an incorrect candidate. Net gain equals recovered minus damaged cases.

Oracle@10 and the number of queries with at least one correct candidate measured candidate-pool coverage. For the primary full-dataset comparison, confidence intervals were reported for accuracy and paired accuracy difference, and McNemar's test assessed paired binary correctness outcomes. These tests support a comparison on this benchmark, not universal generalization.

D. Validation and Test Protocol

The validation and testing protocol was designed to assess the reliability of the procurement benchmarking workflow, reduce potential data leakage, and verify the processing of current-market evidence. During candidate selection, actual_unit_price, absolute_percentage_error, and is_within_10pct were excluded from the selection-time features and reserved for label construction and evaluation. This separation helps prevent the model from exploiting outcome-related information during candidate selection and ensures that evaluation measures are used to assess model performance rather than influence the selection process.

To further minimize textual leakage, explicit currency values and price-related strings were removed from target descriptions before text embedding generation and token-based feature extraction. The latest experimental run identified such strings in 10.8% of the target descriptions, highlighting a potential source of shortcut learning. A clean-subset analysis was subsequently conducted on 9,417 queries without the identified price strings to examine model robustness when this specific source of information was absent. Although this analysis provides an additional check on the effect of explicit price information, it does not establish that all possible leakage pathways have been eliminated. Potential risks associated with duplicated records, correlated metadata, and overlap between training and evaluation data require separate verification.

A separate validation procedure was conducted for the current-market evidence pipeline, which supports benchmark generation using collected market evidence. The test involved 15 evidence records processed through normalization, validation, and benchmark generation. All 15 records passed the implemented validation checks and were included in the benchmark output, resulting in a validation pass rate of 100% (15/15). This result confirms successful data processing for the tested records under the implemented validation rules. However, it does not measure price-prediction accuracy or establish the reliability and representativeness of the generated benchmarks. Further evaluation using larger datasets, independent reference prices, and suitable prediction-error metrics would be required to assess these aspects.

Method	Accuracy@10	Delta vs Rank-1	Recovered	Damaged	Net gain	Override
Rank-1 baseline	37.57%	—	0	0	0	0%
Semantic argmax	38.67%	+1.11 pp	608	491	+117	50.35%
Structured attributes	37.74%	+0.17 pp	Not reported	Not reported	Not reported	Not reported
Semantic + attribute hybrid	38.80%	+1.23 pp	604	474	+130	Not reported
Cross-encoder	≈37.41%	Below baseline	Not reported	Not reported	Not reported	Not reported
Multi-aspect price-aware	38.18%	+0.62 pp	533	468	+65	Not reported
Hand-built gate (comparison run)	38.40%	+0.83 pp	182	94	+88	9.26%
LightGBM binary argmax	38.92%	+1.35 pp	507	364	+143	65.77%
LightGBM LambdaRank argmax	38.38%	+0.81 pp	611	525	+86	59.60%
LightGBM binary + nested-CV gate	38.99%	+1.42 pp	424	274	+150	40.53%

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

VII. RESULTS AND DISCUSSION

A. Candidate Retrieval and Ranking Results

The Rank-1 baseline achieved 37.57% Accuracy@±10%. Oracle@10 was 62.64% (6,615/10,560), indicating that at least one candidate within tolerance was present for 62.64% of queries. For 3,945 queries (37.36%), none of the ten retrieved candidates was correct under the benchmark criterion. Thus, no reranker restricted to the frozen Top-10 pool can exceed this ceiling. There were 2,648 recoverable queries (25.08% of all queries) where Rank-1 was wrong but an alternative candidate was correct.

The primary method achieved 38.99% Accuracy@10, a +1.42 percentage-point gain over Rank-1. Its 95% confidence interval for accuracy was [38.10%, 39.91%], while the paired accuracy-difference interval was [+0.966, +1.932] percentage points. McNemar's test yielded $p = 1.506 \times 10^{-8}$. The primary method recovered 424 Rank-1 errors and damaged 274 previously correct selections, for a net gain of 150.

The raw LightGBM binary argmax achieved 38.92% accuracy and net gain +143, but overrode 65.77% of queries and damaged 364 choices. The nested-CV gate therefore offered the strongest reported accuracy/net-regression trade-off, though it did not lead on every error metric.

B. Price Benchmarking Performance

The reranker returns the original unit price associated with the selected historical candidate; it does not generate a new price. The primary policy reported MAE 6,187.71, MdAPE 23.19% and capped mean absolute percentage error 38.78%. Rank-1 reported MAE 6,729.16 and MdAPE 26.95%. The raw LightGBM binary argmax had a lower MAE of 5,852.28 than the gated method. Therefore, method selection depends on whether the operational priority is tolerance-based candidate correctness, aggregate price error, or conservative overrides. MAE is also sensitive to the price scale and distribution.

C. Semantic Reranking and Evidence Evaluation

Semantic argmax improved Accuracy@10 to 38.67%, but recovered 608 cases while damaging 491. This shows that semantic similarity identifies useful alternatives but is not a reliable standalone proxy for price compatibility. The structured attribute scorer produced a small gain; the semantic–attribute hybrid reached 38.80% but still damaged 474 correct Rank-1 selections. The cross-encoder result was approximately 37.41%, below baseline. The LLM-based approach was not retained as a stable winner because of rate limits and unstable outcomes.

The multi-aspect scorer achieved 38.18%. Error analysis found that recovered alternatives had lower semantic similarity and weaker price consensus than damaged alternatives, suggesting that these signals alone do not reliably distinguish safe replacements. In the learned model, the top gain-based features were semantic-weighted price density in log-price space (dens_w_h10, 29.93%), absolute log-price distance from Rank-1 (abs_lp_vs_rank1, 8.25%), and two weighted price-support features (7.46% and 7.07%). Rank-1 semantic similarity, log price and retrieval-score features also contributed. This describes model contribution, not causality; ablation is needed to quantify feature-group effects.

D. Uncertainty Estimation Results

The historical reranking experiments did not validate a calibrated prediction interval for an individual procurement price. The confidence intervals above describe uncertainty in aggregate benchmark accuracy, not uncertainty for a single price. A separate current-market API smoke test returned benchmarkUnitPrice ₹319,453.47, expectedRange ₹68,457.48–₹1,490,715.48, reliability MEDIUM, 60 candidates and reported inference time of approximately 0.117 seconds. This is one integration example, not a validated uncertainty-estimation experiment. The broad range should be interpreted as an uncertainty indicator, not a guaranteed market price. Before making statistical claims, evaluate interval coverage and width on held-out observations and calibrate reliability labels against empirical errors.

E. Ablation Studies and Comparative Analysis

The study compared several model families and gating policies, but it did not yet complete a controlled feature-group ablation of the final LightGBM model. The clean-subset analysis provides a robustness comparison. Among 9,417 queries without explicit target-text price strings, Rank-1 achieved 38.36% Accuracy@10 and the primary nested-CV gate achieved 39.70% (+1.35 pp). The primary method recovered 374 cases and damaged 247, for a net gain of 127; override rate was 40.31%, MdAPE 22.24%, and the 95% confidence interval for accuracy was [38.70%, 40.68%].

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510112

Future ablations should remove semantic, attribute, price-density/support and retrieval-score features separately. Retrieval coverage should also be measured at K = 5, 10, 20, 30, 50 and 60. A larger K is useful diagnostically, but the intended architecture should prioritize a compact, high-recall pool.

Clean-subset method	Accuracy@10	Delta	Recovered / damaged	Net
Rank-1	38.36%	—	—	—
Semantic argmax	39.41%	+1.05 pp	530 / 431	+99
LightGBM binary argmax	39.65%	+1.30 pp	449 / 327	+122
LightGBM LambdaRank	38.94%	+0.58 pp	545 / 490	+55
LightGBM binary + nested-CV gate	39.70%	+1.35 pp	374 / 247	+127

F. Error Analysis and Limitations

The principal limitation is the candidate-pool ceiling: 37.36% of queries have no correct candidate in Top-10. The primary accuracy improvement is 1.42 percentage points, and absolute Accuracy@10 remains under 40%; results should be reported transparently without claiming state-of-the-art performance. Candidate-side quantity, unit of measure, vendor, award date, location, contract type and structured specifications are unavailable, limiting product comparability. Historical prices can differ from current market prices due to time, location, quantity, supplier and contract conditions. Other limitations include the need for an independent leakage audit, the absence of complete feature ablations, and the lack of a calibration study for the current-market API range/reliability output. The documented API stores jobs in memory and loses them on restart, so it should be treated as an integration prototype unless persistence and production safeguards are added.

Test ID	Input / Action	Expected Result	Actual Result	Status
B01	Evaluate the frozen LambdaMART baseline using the existing evaluation results.	Measure Rank-1 Accuracy@10%.	Recorded result: 37.57%.	Component-level evaluation completed.
B02	Compare the semantic + structured hybrid reranker with the baseline.	Measure the change in ranking accuracy.	Recorded result: 38.80% Accuracy@10%.	Research evaluation recorded.
B03	Evaluate the LightGBM reranker using its recorded results.	Compare ranking performance with the baseline.	Recorded Rank-1 Accuracy@10%: 38.99%.	Research evaluation recorded;
B04	Evaluate top-ranked candidate metrics.	Record Rank-1, Top-3, Top-5 and Top-10 metrics.	37.58%, 48.49%, 54.14%and 62.65%,respectively.	Existing evaluation results.
B05	Run the standalone market-evidence validation pipeline.	Validate and produce benchmark records.	15 raw, 15 normalized, 15 validation, and 15 benchmark records.	Standalone pipelinetest passed.

VIII. CONCLUSION AND FUTURE WORK

This study formulated historical procurement price selection as a candidate reranking task in the CivicEngage NGO tender-support workflow. On a frozen benchmark of 10,560 queries and 105,600 Top-10 candidate rows, semantic, structured-attribute, hybrid, cross-encoder, LLM-based, multi-aspect and learned reranking methods were compared. The results indicate that semantic similarity and historical price consensus are useful evidence, but neither is sufficient on its own to identify the most price-compatible candidate.

The strongest reported policy was a LightGBM binary classifier with a nested cross-validation Rank-1-aware gate. It achieved 38.99% Accuracy@±10%, compared with 37.57% for Rank-1, a gain of 1.42 percentage points. It recovered 424 previously incorrect selections and damaged 274 previously correct selections, producing a net gain of 150. The paired accuracy-difference 95% confidence interval was [+0.966, +1.932] percentage points, with McNemar's p = 1.506

DOI: 10.17148/IJARCCE.2026.1510112

$\times 10^{-8}$. On the clean subset of 9,417 queries without explicit target-text price strings, it achieved 39.70% compared with 38.36% for Rank-1, a gain of 1.35 percentage points.

Feature-importance analysis highlighted semantic-weighted price density, price distance from Rank-1 and weighted price support. The Oracle@10 ceiling of 62.64%, together with 37.36% of queries having no correct candidate in Top-10, shows that upstream retrieval remains a critical bottleneck. The component should be positioned as historical procurement decision support, not as a model that guarantees or directly predicts a current price. Historical candidate prices should be presented separately from current-market evidence.

A. Future Work

First, improve retrieval so a compact candidate pool contains more price-compatible, product-equivalent records. Measure Oracle/Recall@K at several candidate-pool sizes and use larger K as a diagnostic rather than simply passing more candidates downstream. Second, run controlled ablations of semantic, attribute, price-density/support and retrieval features. Third, add reliable candidate-side quantity, unit-of-measure, date, vendor/location and structured specification metadata where available.

Fourth, report performance by commodity family and procurement context. Fifth, perform an independent leakage audit covering feature construction, preprocessing and fold boundaries. Sixth, evaluate the current-market expected range using held-out interval coverage, interval width and calibration metrics; reliability labels should be linked to measured outcomes. Finally, harden the API with persistent job storage, timeouts, structured logs, model warm-up and integration tests for schema validation, rank completeness, exact price preservation, polling and error handling.

In summary, the results provide evidence that a learned Rank-1-aware policy can improve historical candidate selection on the frozen benchmark. Its practical value depends on stronger retrieval, improved product/unit comparability, rigorous uncertainty evaluation, and clear separation of historical and current-market evidence.

REFERENCES

[1]. H. Herrmann and M. J. Cobo Martín, “Bidding for B2B or B2G tenders: toward the adoption of pricing models in practice,” Management Review Quarterly, vol. 75, pp. 1495–1533, 2025. doi: 10.1007/s11301-024-00424-1.

[2]. M. J. García Rodríguez, V. Rodríguez Montequín, F. Ortega Fernández, and J. M. Villanueva Balsera, “Public Procurement Announcements in Spain: Regulations, Data Analysis, and Award Price Estimator Using Machine Learning,” Complexity, vol. 2019, Article 2360610, 2019. doi: 10.1155/2019/2360610.

[3]. D. Balkan and G. A. Akyuz, “Artificial intelligence (AI) and machine learning (ML) in procurement and purchasing decision-support (DS): a taxonomic literature review and research opportunities,” Artificial Intelligence Review, vol. 58, article 341, 2025. doi: 10.1007/s10462-025-11336-1.

[4]. A. Kuchkorov et al., “Machine Learning-based Price Prediction and Analysis of Laptop Specifications,” in 2024 International Conference on Information Science and Communications Technologies (ICISCT), 2024. doi: 10.1109/ICISCT64202.2024.10957051.

[5]. Q. Ruan, D. Shi, and T. Bauernhansl, “Fine-tuning large language models with contrastive margin ranking loss for selective entity matching in product data integration,” Advanced Engineering Informatics, 2025, Art. no. 103538. https://doi.org/10.1016/j.aei.2025.103538.

[6]. R. Berraida and E. Laila, “The Use of Artificial Intelligence to Calculate the Estimate of a Public Procurement Act,” Engineering Proceedings, vol. 97, no. 1, Art. no. 7, 2025. https://doi.org/10.3390/engproc2025097007.

[7]. N. Nananukul, K. Sisaengsuwanchai, and M. Kejriwal, “Cost-efficient prompt engineering for unsupervised entity resolution in the product matching domain,” Discover Artificial Intelligence, vol. 4, Art. no. 56, 2024. https://doi.org/10.1007/s44163-024-00159-8.

[8]. T. Wang, H. Lin, X. Chen, X. Han, H. Wang, Z. Zeng, and L. Sun, “Match, Compare, or Select? An Investigation of Large Language Models for Entity Matching,” arXiv preprint arXiv:2405.16884, 2024.

[9]. R. Joeres, D. B. Blumenthal, and O. V. Kalinina, “Data splitting to avoid information leakage with DataSAIL,” Nature Communications, vol. 16, Art. no. 3337, 2025. https://doi.org/10.1038/s41467-025-58606-8.

[10]. Z. Zhang, W. Zeng, J. Tang, H. Huang, and X. Zhao, “Active in-context learning for cross-domain entity resolution,” Information Fusion, vol. 117, Art. no. 102816, 2025. https://doi.org/10.1016/j.inffus.2024.102816.

[11]. M. Fazekas, B. Tóth, J. Wachs, and A. Abdou, “Public procurement cartels: A large-sample testing of screens using machine learning,” International Journal of Industrial Organization, vol. 104, Art. no. 103228, 2026. https://doi.org/10.1016/j.ijindorg.2025.103228.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.1510112

[12]. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," Advances in Neural Information Processing Systems 30 (NeurIPS), 2017. https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

[13]. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," Proceedings of EMNLP-IJCNLP, pp. 3982–3992, 2019. doi:10.18653/v1/D19-1410. https://aclanthology.org/D19-1410/

[14]. C. J. C. Burges, "From RankNet to LambdaRank to LambdaMART: An Overview," Microsoft Research Technical Report MSR-TR-2010-82, 2010. https://www.microsoft.com/en-us/research/publication/from-ranknet-to-lambdarank-to-lambdamart-an-overview/

[15]. Q. McNemar, "Note on the Sampling Error of the Difference between Correlated Proportions or Percentages," Psychometrika, vol. 12, pp. 153–157, 1947.

[16]. B. Efron and R. J. Tibshirani, An Introduction to the Bootstrap. Boca Raton, FL, USA: Chapman & Hall/CRC, 1993.

 This work is licensed under a Creative Commons Attribution 4.0 International License