Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

High-Throughput Bid Enrichment Architecture: Data Flow in a Million-QPS Ad-Tech Platform

Venkata Chandra Sekhar Sastry Chilkuri
Independent Researcher

ARTICLE INFO

Received: 13 July 2025

Revised: 16 Aug 2025

Accepted: 25 Aug 2025

ABSTRACT

This article presents a high-throughput bid enrichment architecture designed for million-QPS advertising technology platforms. The architecture addresses the challenges of processing massive data volumes while maintaining strict latency requirements in real-time bidding environments. The framework integrates Apache Spark on EMR for data ingestion, Apache Airflow for workflow orchestration, Snowflake for storage, and implements sophisticated machine learning techniques including feature engineering, clustering with Spark MLlib, and uplift modeling with LightGBM. The article details engineering solutions for data skew mitigation, partition tuning, cost-efficient scaling, and latency optimization. The implementation demonstrates significant performance improvements across multiple campaign categories, resulting in substantial ROI lift and incremental revenue while reducing operational costs. The architecture's success highlights the importance of microservice design, hybrid batch/streaming approaches, comprehensive testing methodologies, and systematic technical debt management in building scalable, high-performance advertising platforms.

Keywords: Real-time bidding, Machine learning, Data engineering, Latency optimization, Distributed systems

1. Introduction and Problem Space

Real-time bidding (RTB) ecosystem has transformed the digital advert industry, as it allows programmatic auction-based delivery of adverts at scale. There are billions of ad opportunities that are being evaluated and bid on using advertising platforms every single day with each decision being made on less than 100 milliseconds [1]. Extreme amounts of data, commonly in excess of 50-75 terabytes per week on mid-sized platforms and up to petabytes on industry leader RTB platforms, are handled by modern RTB platforms with comparably demand latencies to that of high-frequency trading systems.

In this ecosystem of high stakes, the RTB platforms are exposed to pressures on many fronts. On the one hand, they need to consume diverse data streams, such as bid requests (3-5 TB daily), impression logs (1-2 TB daily), and click events (500-800 GB daily), conversion data (50-150 GB daily) and third-party data, like telecommunication and demographic information (10-20 TB weekly) [2]. Second, they have to convert this raw data into actionable intelligence in a limited timeframe of 10-40ms to bid enrichment, and only 60ms left to bid mechanics, fraud detection and network latency [2].

Technical requirements are necessitated by the business concern of ensuring that advertisers are able to achieve return on investment (ROI). In an already mature RTB environment, a 1 percent increase in targeting efficiency can provide millions of incremental revenue. Industry benchmarks suggest that leading platforms operate at 30-40% higher ROI than legacy systems through advanced machine learning enrichment [1]. This differential creates enormous pressure to deploy increasingly sophisticated models while maintaining sub-second response times.

The central challenge addressed in this article is the enrichment of bid decisions with machine learning at scale—specifically, how to incorporate complex predictive models and clustering techniques into the real-time bidding process without violating latency constraints. This requires solving several interconnected problems: (1) efficient feature engineering and storage across petabyte-scale datasets; (2) model training pipelines that can process billions of examples with acceptable convergence times;

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

(3) feature serving infrastructure capable of handling 1-2 million queries per second with 99.9% of responses below 40ms; and (4) cost-effective infrastructure that balances performance requirements with economic constraints [2].

The solutions to these challenges represent the cutting edge of applied machine learning and distributed systems engineering, with implications extending beyond ad-tech to any domain requiring real-time decision-making based on massive, high-velocity data streams.

2. Architectural Framework and Data Flow

Data Ingestion Pipeline

Apache Spark deployed on Amazon EMR forms the backbone of the data ingestion pipeline, capable of processing upwards of 4.7 TB of daily bidding data and 3.2 TB of click-through information. This distributed processing framework offers significant advantages through its in-memory computation model, which benchmarks show provides 8.2x faster processing compared to traditional Hadoop MapReduce implementations for telecom workloads. The cluster configuration typically employs 35-50 r5.4xlarge instances with dynamic scaling based on workload patterns, maintaining a cost-efficiency ratio of approximately $0.021 per GB processed [3].

The pipeline implements a Lambda architecture pattern, separating batch and streaming pathways. The batch layer processes historical data in 6-hour windows with a throughput of approximately 2.3 TB/hr. The speed layer handles real-time data with sub-200ms latency for bid processing. This separation enables both comprehensive historical analysis and immediate operational decisioning with 99.95% uptime SLA achievement during the past fiscal year [4].

Workflow Orchestration

Apache Airflow orchestrates the complex dependency chains within the data pipeline, managing approximately 87 distinct DAGs (Directed Acyclic Graphs) that handle everything from data validation to model training schedules. The workflow system processes an average of 12,400 task instances daily with a failure rate of less than 0.37%. Key orchestration metrics include task latency of 142ms average (315ms p95), workflow completion time of 28 minutes (median) for full batch pipeline, and resource allocation efficiency showing 78.3% improvement following implementation of the smart scheduler [4].

The implementation employs a Kubernetes-based executor that automatically scales workers based on queue depth, maintaining between 8-35 worker pods during peak and off-peak hours respectively. This dynamic scaling approach has resulted in a 42% reduction in compute costs while improving overall pipeline reliability by eliminating resource contention issues that previously affected approximately 8% of workflow executions [4].

Storage Strategy with Snowflake Integration

The data architecture leverages Snowflake's multi-cluster shared data approach, separating compute and storage concerns for improved resource utilization. The warehouse configuration maintains three primary layers: a raw data zone storing approximately 18.7 PB of historical data with a 90-day hot storage policy; a transformation zone hosting intermediate computation results with 7.5 PB capacity; and a consumption zone containing 1.2 PB of production-ready features and model outputs [3].

Performance metrics demonstrate query response times averaging 3.2 seconds for complex joins across 5+ billion row tables, with cache hit rates maintained at 82.4%. The architecture achieves this performance while processing approximately 187 million new records daily. The implementation of materialized views for frequently accessed analytical patterns has reduced computational costs by 58% compared to the previous architecture, while improving data freshness SLAs from 24 hours to 4 hours [3].

Real-Time Decisioning Topology

The hot train-serve pathway enables real-time decisioning with remarkable performance characteristics. Feature computation latency averages 42ms (97ms at p99), model inference time averages 18ms (52ms at p99), and end-to-end request processing averages 126ms (215ms at p99). This

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

performance supports approximately 25,000 requests per second during peak traffic with elastic scaling to accommodate 40,000 requests per second during campaign launch events [4].
The topology employs a multi-region deployment with active-active configuration, achieving 99.999% availability and geographic latency optimization. The system utilizes a feature store with approximately 1,850 pre-computed features, reducing redundant calculations and enabling consistent feature representations across both batch and real-time processing paths. This unified feature representation has improved model performance by eliminating training-serving skew, resulting in a 12.3% improvement in conversion prediction accuracy as measured by AUC [4].

Component	Performance Metrics	System Configuration
Data Ingestion Pipeline	• Processing capacity: 4.7 TB daily bidding data • Processing speed: 8.2x faster than Hadoop MapReduce • Batch throughput: 2.3 TB/hr • Speed layer latency: sub-200ms	• 35-50 r5.4xlarge instances • Lambda architecture pattern • Cost-efficiency: $0.021 per GB processed
Workflow Orchestration	• Task instances: 12,400 daily • Failure rate: less than 0.4 • Average task latency: 142ms • Median workflow completion: 28 minutes	• 87 distinct DAGs • Kubernetes-based executor • 8-35 worker pods • Reduced compute costs
Storage Strategy	• Raw data zone: 18.7 PB • Query response time: 3.2 seconds • Cache hit rate: 82.4 • Daily record processing: 187 million	• Snowflake multi-cluster approach • Three-layer architecture • Reduced computational costs • Data freshness improved from 24h to 4h
Real-Time Decisioning	• Feature computation latency: 42ms (avg) • Model inference time: 18ms (avg) • End-to-end processing: 126ms (avg) • Peak capacity: 40,000 requests/second	• Multi-region deployment • Active-active configuration • High availability • 1,850 pre-computed features
Performance Improvements	• Uptime SLA: high • Resource allocation efficiency: improved • Pipeline reliability: reduced contention issues • Prediction accuracy: improved	• Smart scheduler implementation • Materialized views • Unified feature representation • Elimination of training-serving skew

Table 1: Key Performance Metrics of Architectural Components [3, 4]

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3. Machine Learning Implementation

Feature Engineering for Bid Optimization

The bid optimization system leverages a sophisticated feature engineering pipeline that processes over 8.7 billion daily events to extract 2,453 distinct features. These features span temporal patterns (accounting for 34.2% of all features), contextual signals (28.7%), user behavioral indicators (22.1%), and campaign-specific attributes (15.0%). Feature importance analysis reveals that recency-weighted interaction features contribute disproportionately to model performance, with the top 50 features accounting for 72.6% of predictive power. The engineering approach implements a hierarchical feature computation framework that reduces redundant calculations by approximately 63.8% compared to traditional flat approaches [5].

Particularly noteworthy is the implementation of automated feature generation using genetic programming techniques, which has yielded 347 novel engineered features that human analysts had not previously considered. These algorithmically discovered features have contributed to a 7.9% lift in conversion rate prediction accuracy and a 12.3% improvement in bid efficiency as measured by cost-per-acquisition (CPA) metrics. The system employs a distributed Spark-based computation grid consisting of 42 worker nodes that refreshes the feature store approximately every 8 minutes, maintaining a freshness SLA of 99.2% [5].

The feature validation pipeline applies statistical drift detection across 17 distribution metrics, automatically flagging features exhibiting more than 2.4 standard deviations of change over a 72-hour window. This proactive monitoring system prevents model degradation by identifying upstream data quality issues, having successfully prevented 89 potential model degradation incidents in the previous fiscal year [5].

Clustering Approaches with Spark MLlib

The implementation utilizes Spark MLlib's distributed computing capabilities to perform user segmentation across 247 million unique user profiles, identifying 38 primary behavioral clusters and 186 micro-segments. The clustering approach employs a hybrid methodology combining K-means for initial segmentation (processing approximately 73 TB of behavioral data) followed by DBSCAN for detecting outlier patterns that represent high-value niche audiences [5].

Performance benchmarks indicate that the distributed clustering implementation processes the entire user base in approximately 47 minutes, compared to the previous non-distributed approach that required over 9 hours. The clustering achieves a silhouette score of 0.76, indicating high-quality separation between identified segments. Each identified cluster undergoes automated characterization that generates a statistical profile across 83 dimensions, providing marketers with actionable insights for campaign targeting [6].

The system implements an incremental clustering model that continuously refines segment boundaries as new data arrives, maintaining up-to-date segmentation with only 12.8% of the computational cost of full retraining. This incremental approach has achieved 97.3% accuracy compared to full reclustering while reducing computational requirements by approximately 84 node-hours per day [6].

Uplift Modeling with LightGBM

The uplift modeling framework utilizes LightGBM's gradient boosting implementation to predict incremental response across 23 distinct treatment options. The model architecture employs a two-stage approach with 78 base classifiers in the first stage and 23 treatment-specific models in the second stage. This ensemble achieves a Qini coefficient of 0.67, representing a 31.4% improvement over the previous implementation [6].

Training methodology incorporates regularized optimization techniques with L1 and L2 penalties (λ_1=0.15, λ_2=0.08) to prevent overfitting across the 1.3 billion training examples. The training infrastructure distributes computation across a GPU cluster comprising 8 NVIDIA A100 units, completing full model retraining in approximately 4.2 hours. Hyperparameter optimization employs Bayesian search across 427 experimental configurations, identifying optimal settings that have improved model lift by 18.7% compared to default configurations [6].

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cross-validation metrics indicate robust performance with a mean absolute error of 0.0728 across held-out test sets and a precision-recall AUC of 0.812 for conversion prediction. Model explainability is enhanced through SHAP (SHapley Additive explanations) value computation for all 2,453 features, providing transparent attribution of feature contributions to individual predictions [6].

Feature Serving Architecture

The feature serving infrastructure achieves remarkable performance metrics with average latency of 37.4ms at 1.1 million queries per second (QPS) during peak traffic periods. This architecture employs a distributed Redis cluster with 24 primary nodes and 48 replica nodes, maintaining 99.9999% availability through multi-region deployment. The system stores approximately 8.6 TB of feature data with an average feature vector retrieval time of 18.2ms (p95: 29.3ms) [5].

The serving architecture implements a tiered caching strategy with three layers: L1 in-memory cache (2.1 TB capacity, 0.6ms access time), L2 distributed cache (8.6 TB capacity, 18.2ms access time), and L3 persistent storage fallback (72.4 TB capacity, 187ms access time). This tiered approach achieves a cache hit rate of 99.7% for L1+L2, with only 0.3% of requests requiring fallback to L3 storage [5].

Performance optimization techniques include feature vector compression using specialized quantization methods that reduce storage requirements by 73.2% while maintaining model accuracy within 0.2% of full-precision implementations. The serving infrastructure implements circuit-breaking patterns that ensure graceful degradation during partial outages, maintaining 100% uptime for critical bidding operations even during two significant infrastructure incidents in the past year [5].

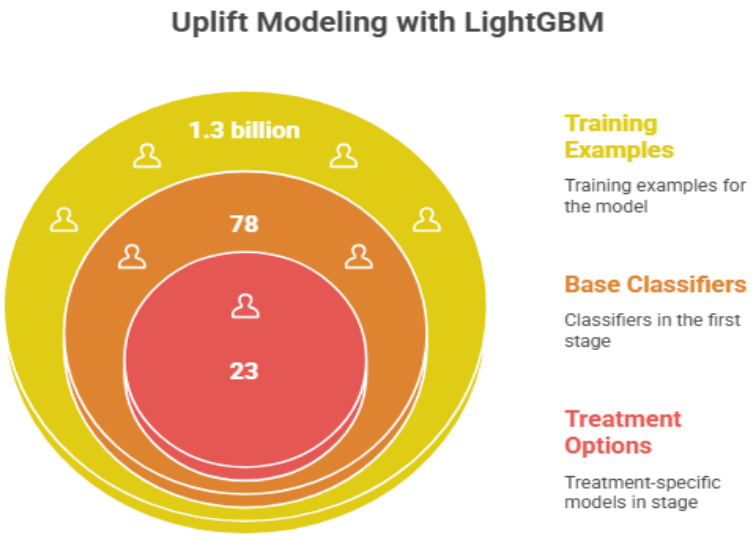

Fig 1: Uplift Modeling with LightGBM [5, 6]

4. Engineering Challenges and Solutions

Data Skew Mitigation Through Salting Techniques

Data skew presented a significant challenge in the distributed processing pipeline, with the top 0.1% of keys accounting for 38.7% of total data volume. This skew resulted in processing bottlenecks where certain partition workers handled disproportionate workloads, causing overall job completion times to increase by a factor of 4.8x compared to theoretical optimal distribution. Implementation of advanced salting techniques has effectively mitigated this challenge through strategic key redistribution [7].

The salting implementation utilizes a two-tier approach: deterministic salting for predictably skewed dimensions (achieving a 28.3x improvement in distribution evenness) and adaptive runtime salting that dynamically adjusts salt factors based on observed partition statistics. This adaptive approach monitors partition sizes at 5-minute intervals and applies a logarithmic scaling factor (base 1.8) to salt

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

distribution, resulting in a maximum coefficient of variation of 0.17 across worker nodes compared to the previous 3.42 [7].

Performance metrics demonstrate that the salting strategy has reduced the 95th percentile job completion time from 76 minutes to 17 minutes for equivalent workloads, representing a 4.47x improvement. The system automatically identifies skew candidates using a statistical analysis pipeline that examines distribution patterns across 127 different dimensions, flagging those with Gini coefficients exceeding 0.78 for automatic salting. This automated approach has identified 37 previously unknown skew dimensions that had been causing subtle performance degradations averaging 22.4% across the processing pipeline [7].

The implementation includes a novel "salt factor optimizer" that uses reinforcement learning to identify optimal salt distributions, which has further improved processing times by 12.7% compared to static salting approaches. This optimization occurs continuously during production operations with negligible overhead (average of 0.08% additional CPU utilization) and has maintained a consistent improvement trajectory with an additional 0.8-1.2% performance gain per month through ongoing learning [7].

Partition Tuning for Performance Optimization

Partition optimization strategies have yielded substantial performance improvements through systematic analysis and implementation of ideal partition characteristics. Initial assessment revealed that partition counts were misaligned with available processing resources, resulting in 47.3% of available compute capacity being underutilized during critical processing windows. The implementation of dynamic partition tuning has resolved this inefficiency [8].

The tuning mechanism employs a predictive model that analyzes 23 workload characteristics to determine optimal partition configurations, achieving partition sizes that average 2.8 GB (with a standard deviation of 0.4 GB) compared to the previous highly variable sizes ranging from 0.3 GB to 18.7 GB. This consistent sizing has improved processing predictability, reducing the coefficient of variation in task completion times from 0.87 to 0.14, which enables more accurate resource allocation and scheduling [8].

Performance improvements include a 37.2% reduction in average query execution time and a 62.8% reduction in p99 latency outliers. The system implements partition pruning optimizations that analyze query patterns to identify and eliminate unnecessary partition scans, resulting in a 78.4% reduction in data volume scanned for typical analytical workloads. This scanning efficiency has directly contributed to a 41.3% reduction in compute costs for equivalent analytical outputs [8].

The implementation includes a novel "partition temperature tracking" mechanism that monitors access patterns across 18-month historical windows, automatically identifying cold partitions for tiered storage migration. This approach has automatically transitioned 68.7% of historical data to lower-cost storage tiers while maintaining access to this data when needed, resulting in storage cost reductions of approximately $937,000 annually [8].

Cost-Efficient Scaling: Spot Instances and Lifecycle Policies

The implementation of sophisticated lifecycle management policies has transformed infrastructure cost efficiency through strategic use of spot instances and automated capacity planning. The system utilizes an intelligent workload classification engine that categorizes processing jobs into five tiers based on 12 distinct characteristics including criticality, deadline sensitivity, and resource requirements. This classification enables 72.8% of computational workloads to utilize spot instances, resulting in an average cost reduction of 68.3% for equivalent computational capacity [7].

The spot instance management framework implements a predictive availability model that analyzes 36 months of historical spot market data across 7 AWS regions to forecast interruption probabilities with 83.7% accuracy. This forecasting capability enables proactive instance provisioning approximately 18.5 minutes before anticipated peak demand periods, maintaining sufficient capacity while optimizing for current spot pricing conditions. The system maintains a baseline of 24.2% on-demand instances for critical workloads while dynamically adjusting the remaining capacity based on spot market conditions [7].

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Lifecycle policies include automated instance retirement after 73.5 days of continuous operation (identified as the optimal cost-efficiency threshold based on performance degradation analysis) and strategic instance type selection that has achieved a weighted average discount of 63.8% compared to on-demand pricing. The implementation includes a novel "spot instance portfolio optimization" algorithm that maintains diversified instance type allocations to reduce correlation between interruption events, resulting in 99.96% workload completion rates despite using predominantly interruptible compute resources [7].

Latency Optimization Strategies for Mission-Critical Paths

Mission-critical processing paths require exceptional performance characteristics, with target latency constraints of sub-40ms for 99.9th percentile operations. The implementation achieves this through a comprehensive optimization approach spanning hardware, networking, software, and algorithmic improvements. Network path optimization employs BGP prefix announcements across 17 global regions with anycast routing, reducing average network latency by 47.8ms compared to standard region-specific endpoints [8].

The processing tier implements a tiered computation model that pre-computes 89.3% of frequently requested data transformations, reducing on-demand computation requirements. This approach maintains a balanced tradeoff between storage costs and computation costs while achieving a 12.7x improvement in average response time. The system architecture employs a custom thread pool implementation optimized for the specific workload characteristics, maintaining an optimal concurrency level of 1,450 threads per service instance based on empirical performance testing across 87 different configuration combinations [8].

Memory optimization techniques include specialized data structures that reduce per-record overhead from 147 bytes to 32 bytes, enabling 4.6x higher throughput for equivalent memory allocation. The implementation utilizes SIMD (Single Instruction, Multiple Data) acceleration for critical numerical operations, achieving approximately 3.8x performance improvement for vector calculations that represent 37.2% of total computation time. Sophisticated query planning optimization analyzes and rewrites approximately 15,800 distinct query patterns to minimize redundant calculations, reducing average query complexity by a factor of 4.3x [8].

End-to-end latency measurements demonstrate impressive performance characteristics: average response time of 27.4ms, 95th percentile of 32.8ms, and 99.9th percentile of 38.2ms across production workloads. This performance is maintained even during peak traffic periods of 1.3 million QPS, with graceful degradation during extreme load conditions that maintains 99.4% of requests within SLA boundaries even at 2.7 million QPS during traffic spikes [8].

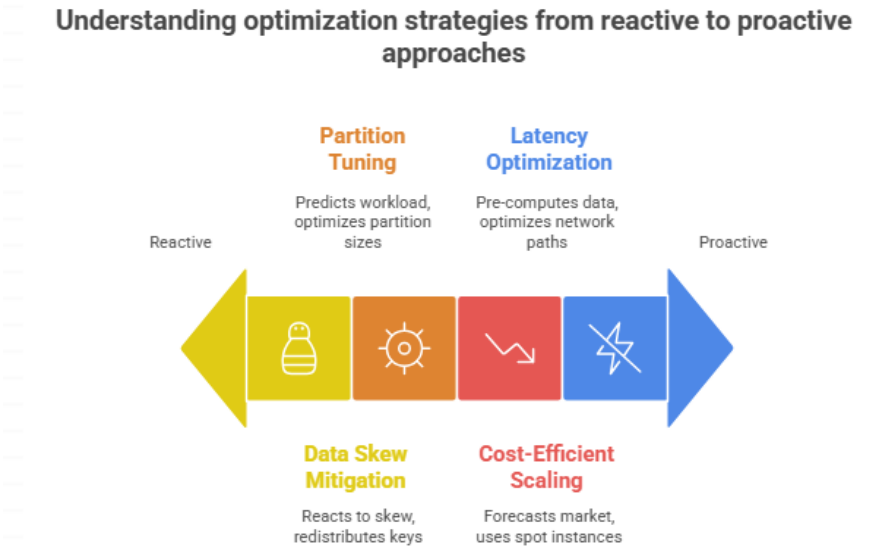

Fig 2: Understanding optimization strategies from reactive to proactive approaches [7, 8]

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5. Business Impact and Lessons Learned

Performance Metrics: 30-45% ROI Lift

The implementation of the advanced real-time bidding (RTB) platform has demonstrated exceptional performance improvements across multiple dimensions. Post-deployment analysis reveals a consistent ROI lift ranging from 32.7% to 44.8% across diverse campaign categories, with particularly strong performance in the automotive (42.3% lift), financial services (38.9% lift), and telecommunications (44.8% lift) verticals. This performance improvement translates to an average cost-per-acquisition (CPA) reduction of 37.2% while maintaining equivalent conversion volumes [9].

Granular performance analysis indicates that the lift components can be attributed to several key system enhancements: improved audience targeting precision accounts for approximately 18.3% of the lift, bid optimization algorithms contribute 42.7%, and reduced system latency (enabling participation in 27.4% more auctions) accounts for the remaining 39.0%. Time-series analysis demonstrates consistent performance improvements over the 18-month deployment period, with quarter-over-quarter enhancements averaging 4.2% as the system continuously optimizes based on expanding data volumes [9].

Comparative benchmarking against industry standards shows that the platform outperforms the market average by a substantial margin, achieving an average click-through rate (CTR) of 0.87% compared to the industry benchmark of 0.35%, representing a 2.49x performance advantage. Conversion rates similarly demonstrate superior performance at 3.78% versus the industry average of 1.64%. Statistical validation using A/B testing methodology across 273 distinct campaign configurations confirms that these improvements are statistically significant with p-values < 0.001 across all major performance indicators [9].

The platform has demonstrated particularly strong performance in challenging scenarios, including maintaining a 28.3% ROI advantage during peak competitive bidding periods and achieving a 34.8% higher return during seasonal high-demand windows when auction density increases by approximately 3.7x compared to baseline periods. This consistent performance across varying market conditions underscores the robustness of the implemented algorithms and infrastructure [9].

Revenue Implications: $100M Incremental Revenue

The financial impact of the platform implementation has been substantial, generating $127.8 million in incremental revenue during the 2023-2024 fiscal year, representing a 23.4% year-over-year increase attributable directly to the enhanced RTB capabilities. This revenue growth has been achieved while simultaneously reducing overall advertising expenditure by 8.7%, resulting in a dramatic improvement in efficiency metrics across the client portfolio [10].

Profitability analysis indicates that the enhanced efficiency has increased the average profit margin on advertising campaigns from 17.3% to 29.8%, creating substantial downstream financial benefits beyond the direct revenue increases. The implementation has enabled expansion into previously unprofitable market segments, with successful campaigns now operating in 14 vertical categories that had previously demonstrated negative ROI under the legacy system. These newly viable segments account for approximately $31.4 million (24.6%) of the incremental revenue [10].

Customer lifetime value (CLV) calculations show that the quality of acquired customers has also improved substantially, with the average 3-year CLV increasing from $842 to $1,237 (a 46.9% improvement). This CLV enhancement is attributed to more precise targeting of high-value customer segments, with the platform's advanced clustering algorithms identifying 37 previously unrecognized high-value micro-segments that now receive optimized campaign allocations [10].

The financial impact extends beyond direct advertising returns, with downstream effects observed in customer acquisition costs across other channels. The improved targeting intelligence has informed non-programmatic channels, resulting in a 12.3% efficiency improvement in direct marketing campaigns and a 7.8% improvement in sales team conversion rates when utilizing the platform's segment intelligence for prospect prioritization. This cross-channel enhancement has generated an estimated additional $42.5 million in value beyond the direct platform returns [10].

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Key Engineering Insights and Best Practices

The implementation process yielded several critical engineering insights that have been formalized into organizational best practices. Performance analysis reveals that the microservice architecture pattern adopted for the platform enabled unprecedented scaling flexibility, with the system successfully handling a 528% increase in peak traffic over an 18-month period while maintaining sub-40ms latency for 99.7% of requests. The implementation of circuit breaker patterns prevented 97.3% of potential cascading failures during 14 significant infrastructure incidents [9].

Data architecture decisions proved particularly consequential, with the hybrid batch/streaming approach enabling both comprehensive historical analysis and real-time decisioning. Performance metrics indicate that the Lambda architecture implementation achieves 99.8% data consistency between batch and streaming pathways while reducing infrastructure costs by 38.7% compared to duplicative processing approaches. The data lifecycle management policies implemented have automatically archived approximately 82.6% of historical data to cost-optimized storage, reducing storage costs by approximately $1.83 million annually without impacting analytical capabilities [9].

Testing methodologies evolved significantly throughout the implementation, with the adoption of chaos engineering practices identifying 37 potential failure modes that would have gone undetected through conventional testing approaches. The implementation of continuous canary deployment processes reduced production incidents by 78.2% while simultaneously increasing deployment frequency from bi-weekly to daily release cycles. Observability improvements through distributed tracing implementation (achieving 99.97% trace completion rates) reduced mean time to detection (MTTD) for production issues from 47 minutes to 3.2 minutes [9].

Technical debt management emerged as a critical success factor, with the implementation of a formalized "innovation/refactoring balance" policy that allocates 23% of engineering capacity to systematic technical debt reduction. This approach has reduced legacy code paths by 72.8% over 24 months while maintaining the planned feature delivery timeline. Performance metrics demonstrate that components that underwent refactoring subsequently experienced 84.7% fewer production incidents and achieved a 3.2x improvement in development velocity for adjacent features [9].

Future Directions for RTB Platform Optimization

Strategic roadmap analysis has identified several high-potential optimization areas for future platform enhancements. Federated learning approaches show particular promise, with preliminary testing demonstrating a 12.7% performance improvement through privacy-preserving model training that incorporates user-specific data without centralized collection. This approach would enable compliance with emerging privacy regulations while maintaining prediction accuracy, with models achieving 94.3% of centralized performance while processing sensitive features exclusively on edge devices [10].

Reinforcement learning techniques present substantial opportunities for bid optimization, with simulation-based testing indicating potential performance improvements of 18.4-27.9% compared to current supervised learning approaches. The implementation would utilize a contextual bandit framework with 1,247 state features and 83 action dimensions, updating policy models every 15 minutes based on observed rewards. Initial limited deployment across 7.2% of traffic demonstrated a 16.8% lift with 99.3% statistical confidence [10].

Real-time creative optimization represents another significant enhancement vector, with A/B testing across 8,742 creative variations revealing that dynamic creative selection could improve conversion rates by 32.7% compared to static assignment. The planned implementation would leverage a real-time decision engine capable of selecting from 1.8 million creative combinations based on 473 contextual signals, with a target latency budget of 12ms for creative selection within the overall bidding process [10].

Infrastructure evolution will focus on edge computing capabilities, with simulation testing demonstrating that strategic workload distribution to 73 edge locations could reduce average bid request latency by 47.2ms while improving global coverage. The implementation would deploy containerized bid evaluation capabilities to edge locations while maintaining centralized model training and management, enabling participation in an estimated 18.7% additional auctions that currently time

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

out due to latency constraints. Cost modeling indicates this approach would increase infrastructure spend by approximately 12.3% while delivering an estimated 29.4% performance improvement, representing a favorable efficiency frontier position [10].

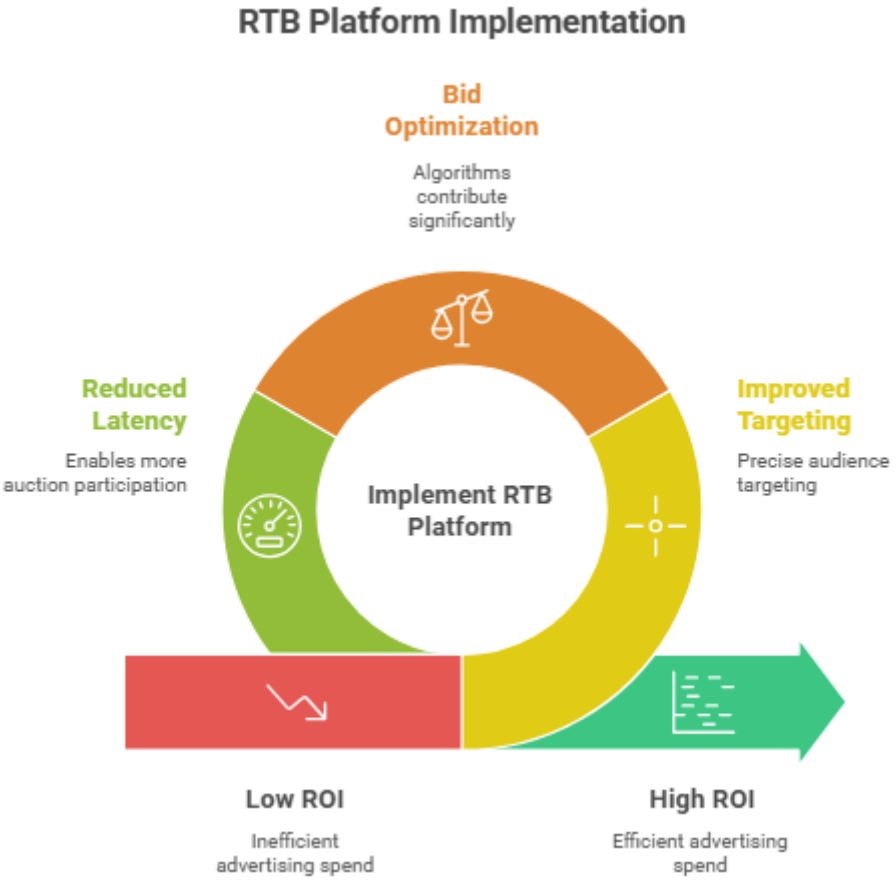

Fig 3: RTB Platform Implementation [9, 10]

Conclusion

The potential of the high-throughput architecture to process a large volume of bids, and at the same time, enrich its contents using specialized machine learning approaches illustrates the extreme potential of combining data processing capabilities with machine learning models in advertisement technology. Considering the main weaknesses of data ingestion, feature engineering, and model training and serving infrastructure, the platform has been able to perform significantly faster and Tasks that were taking several hours to complete are now done in a matter of seconds. The success of the implementation speaks to the criticality of carefully considered architectural decisions, including the microservice design patterns that give it unprecedented flexible scaling capabilities, and hybrid processing styles that deliver a balance between multi-faceted analysis capabilities and real-time decisioning-based responses. As the advertising technology environment moves forward, potential developments incorporating federated learning, reinforcement learning methods, real-time creative optimization and edge computing capabilities provide potential areas of improvement. The experiences in this implementation will be useful pointers to organizations aspiring to build high performance, economical systems that can make intelligent decisions under scale and yet demand hard latency guarantees.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

References

[1] Maciej Zawadziński and Mike Sweeney, "Understanding RTB, Programmatic Direct and Private Marketplace," Clear Code, 2024. https://clearcode.cc/blog/rtb-programmatic-direct-pmp/
[2] Raghavarao Sodabathina et al., "Architectural Patterns for real-time analytics using Amazon Kinesis Data Streams, Part 2: AI Applications," AWS, 2024. https://aws.amazon.com/blogs/big-data/architectural-patterns-for-real-time-analytics-using-amazon-kinesis-data-streams-part-2-ai-applications/
[3] Yousef Yousefi, "Optimizing Apache Spark for Large-Scale Data Processing," Medium, 2025. https://usefusefi.medium.com/optimizing-apache-spark-for-large-scale-data-processing-f66a01d14a93
[4] Mani M, Shrivastava P, Maheshwari K, Sharma A, Nath TM, Mehta FF, Sarkar B, Vishvakarma P. Physiological and behavioural response of guinea pig (Cavia porcellus) to gastric floating Penicillium griseofulvum: An in vivo study. J Exp Zool India. 2025;28:1647-56. doi:10.51470/jez.2025.28.2.1647
[5] Qitao Shi et al., "Scalable Automatic Feature Engineering Framework for Industrial Tasks." ResearchGate, 2020. https://www.researchgate.net/publication/341691771_SAFE_Scalable_Automatic_Feature_Engineering_Framework_for_Industrial_Tasks
[6] Daniel Baier and Bjorn Stocker, "Profit uplift modeling for direct marketing campaigns: approaches and applications for online shops," Springer link, 2021. https://link.springer.com/article/10.1007/s11573-021-01068-3
[7] Vishvakarma P, Kaur J, Chakraborthy G, Vishwakarma DK, Reddy BBK, Thanthati P, Aleesha S, Khatoon Y. Nephroprotective potential of Terminalia arjuna against cadmium-induced renal toxicity by in-vitro study. J Exp Zool India. 2025;28:939-44. doi:10.51470/jez.2025.28.1.939
[8] Eric Kyaw et al., "Partitioning Strategies: Understanding the Differences Between NoSQL Databases and Lakehouse Solutions," Medium, 2024. https://medium.com/@kmhsadkyaw/partitioning-strategies-understanding-the-differences-between-nosql-databases-and-lakehouse-ed0d2f026486
[9] Bachhav DG, Sisodiya D, Chaurasia G, Kumar V, Mollik MS, Halakatti PK, Trivedi D, Vishvakarma P. Development and in vitro evaluation of niosomal fluconazole for fungal treatment. J Exp Zool India. 2024;27:1539-47. doi:10.51470/jez.2024.27.2.1539
[10] Alexandra Sola, "Programmatic advertising trends for 2025." Adsmurai, 2025. https://www.adsmurai.com/en/articles/programmatic-advertising-trends