TREND ANALYTICS AT SCALE

Team – Spark Syndicate

Kashinath Alias Kapil Subhash Naik *kashinathn@iisc.ac.in*

Sujith Shetty *sujith1@iisc.ac.in*

Atreyee Mondal *atreyeem@iisc.ac.in*

Pranav N *pranavn@iisc.ac.in*

Problem & Motivation

- **Volume Overload:** E-commerce platforms receive 10,000+ reviews/minute[1]. Manual monitoring is impossible.
- **Buried Insights:** Critical product defects and sentiment shifts are lost in unstructured data noise.
- **Cost of Delay:** More than 75% of consumers read reviews[2]. A 24-hour lag in detecting a defect can ruin a brand's reputation.
- **Need for Speed:** Traditional batch processing is too slow. Real-time detection is critical for rapid response.

1: https://www.aboutamazon.com/news/policy-news-views/amazon-fake-reviews-fraud-impact-report-2023
2: https://www.demandsage.com/online-review-statistics/

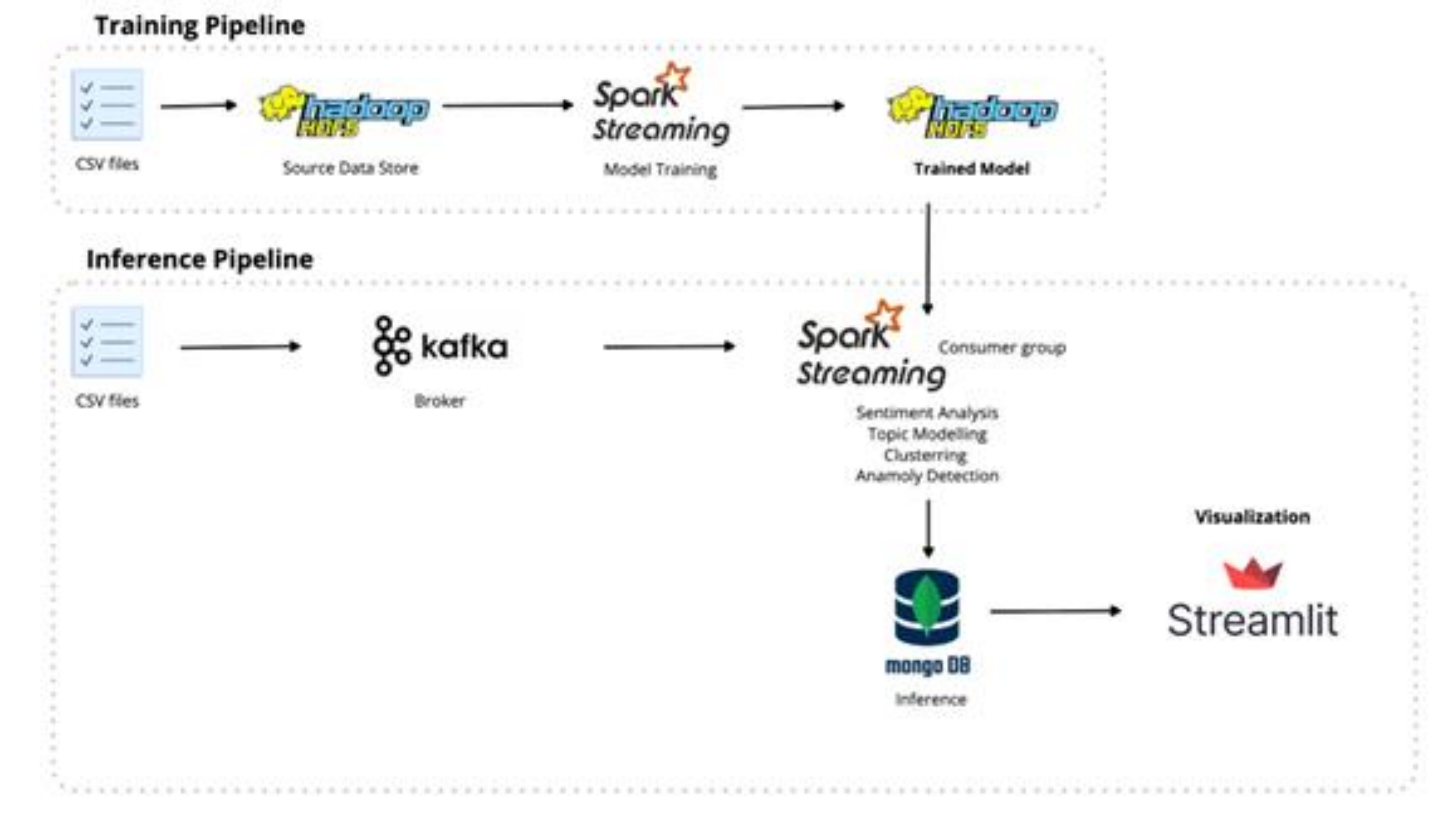
System Design Goals

Low Latency - Sub-minute processing from review ingestion to dashboard visualization for immediate awareness.

High Throughput - Capable of handling multiple reviews per minute from multiple sources with horizontal scalability to meet demand.

Fault Tolerance - Automatic recovery from node failures with zero data loss, ensuring consistent system reliability.

System Architecture



Model Training (Batch)

Sentiment Analysis: Logistic Regression trained on 153k records. Uses TF-IDF and N-grams.

Topic Modeling (LDA): Unsupervised learning to discover hidden themes (e.g., "Quality," "Delivery," "Price").

Clustering: K-Means to group reviews into segments like "Critical Issues" or "Satisfied Customers."

Stream Processing Pipeline

01

1. Ingest

Kafka topic ingestion in 10-second micro-batches.

02

2. Preprocess

Tokenization, Stopword removal, Vectorization (50k vocab).

03

3. ML Inference

Apply Sentiment Classifier & Topic Modeling (LDA).

04

4. Store

Write enriched JSON documents to MongoDB.

Sentiment Analysis

Model: Logistic Regression

Training Corpus: 153k Customer Review Records

(70% Train, 15% Validation and 15% Test)

Vectorization: TF –IDF

Context Capture: N-grams

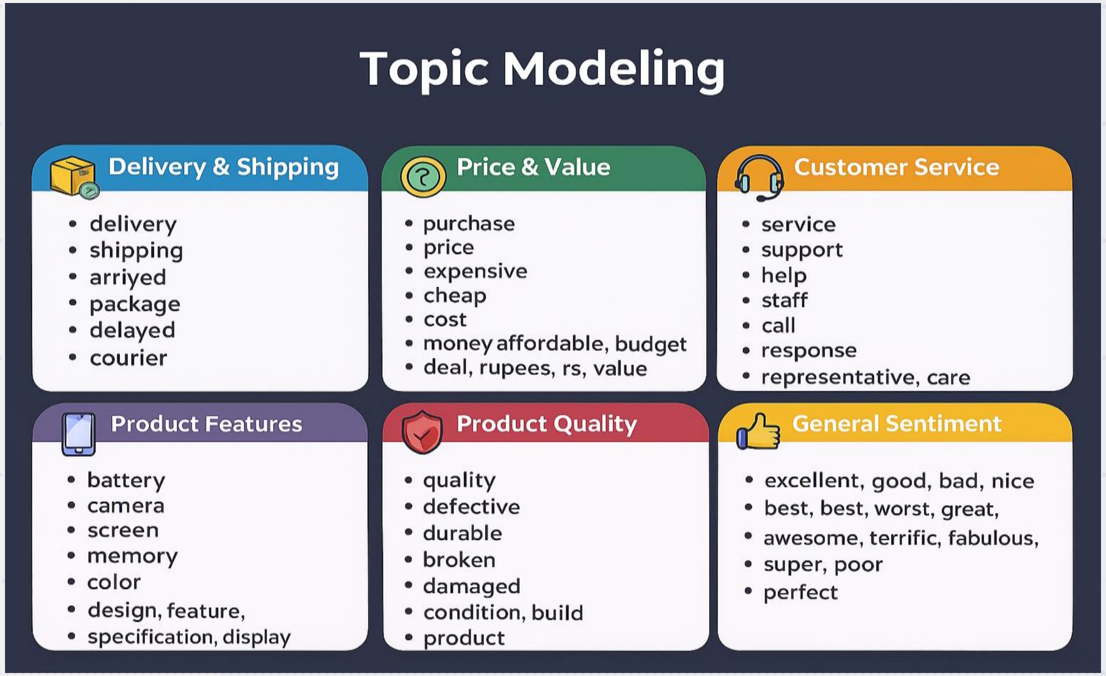
Metric: F1 Score ~0.961

Latent Dirichlet Allocation (LDA)

Mechanism: Iterative learning of topic-word distributions (20 iterations, 5 topics).

Input: 5,000-word vocabulary vectorization.

Metric: Log Perplexity



Topic Modeling

Delivery & Shipping

- delivery
- shipping
- arriyed
- package
- delayed
- courier

Price & Value

- purchase
- price
- expensive
- cheap
- cost
- money affordable, budget
- deal, rupees, rs, value

Customer Service

- service
- support
- help
- staff
- call
- response
- representative, care

Product Features

- battery
- camera
- screen
- memory
- color
- design, feature,
- specification, display

Product Quality

- quality
- defective
- durable
- broken
- damaged
- condition, build
- product

General Sentiment

- excellent, good, bad, nice
- best, best, worst, great,
- awesome, terrific, fabulous,
- super, poor
- perfect

K-Means Clustering

Features: Sentiment, Star Ratings, Text Length, Topic Dist.

Optimization: Silhouette Score determines optimal *k*.

Labels: Auto-generated (e.g., "Critical Issues", "Satisfied").

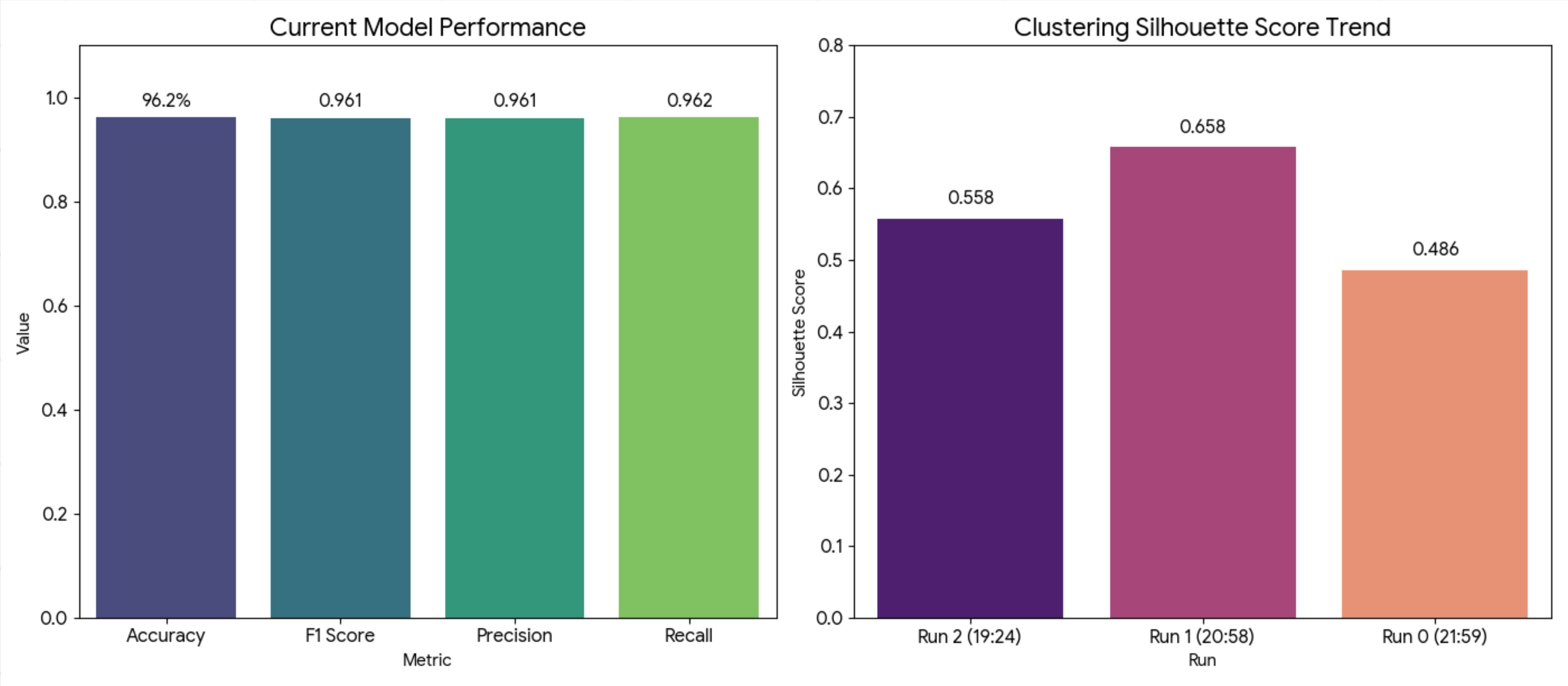
Real-Time Anomaly Detection

Negative Rate Threshold – 70%

Detection Window – 10s

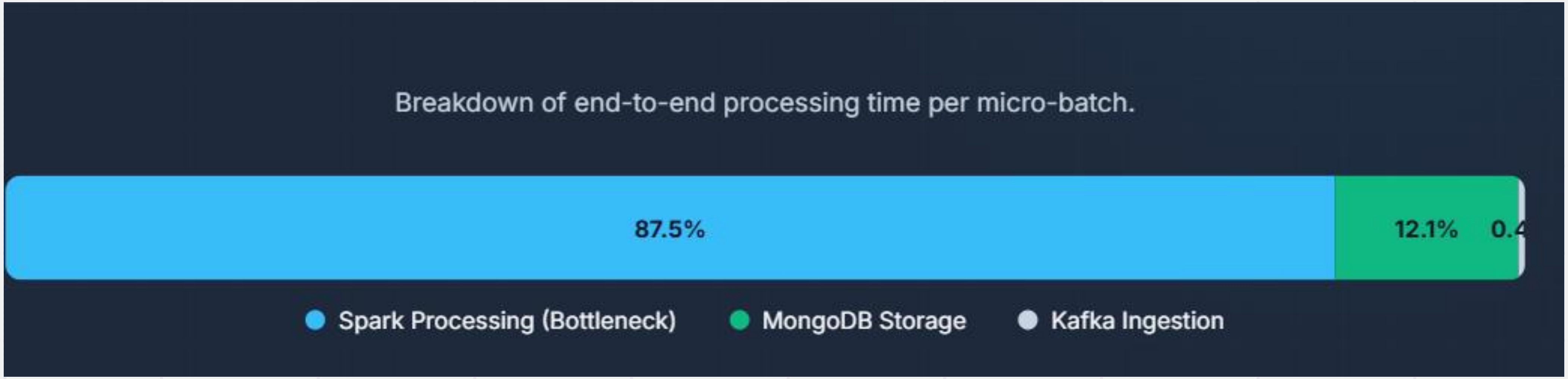
Trigger: Alert raised if ratio > 0.70 in a micro-batch.

Evaluation I: ML Performance



Insight: The model is highly reliable for identifying satisfied customers (95% F1) and balanced in detecting critical issues, meeting the primary goal of trend detection.

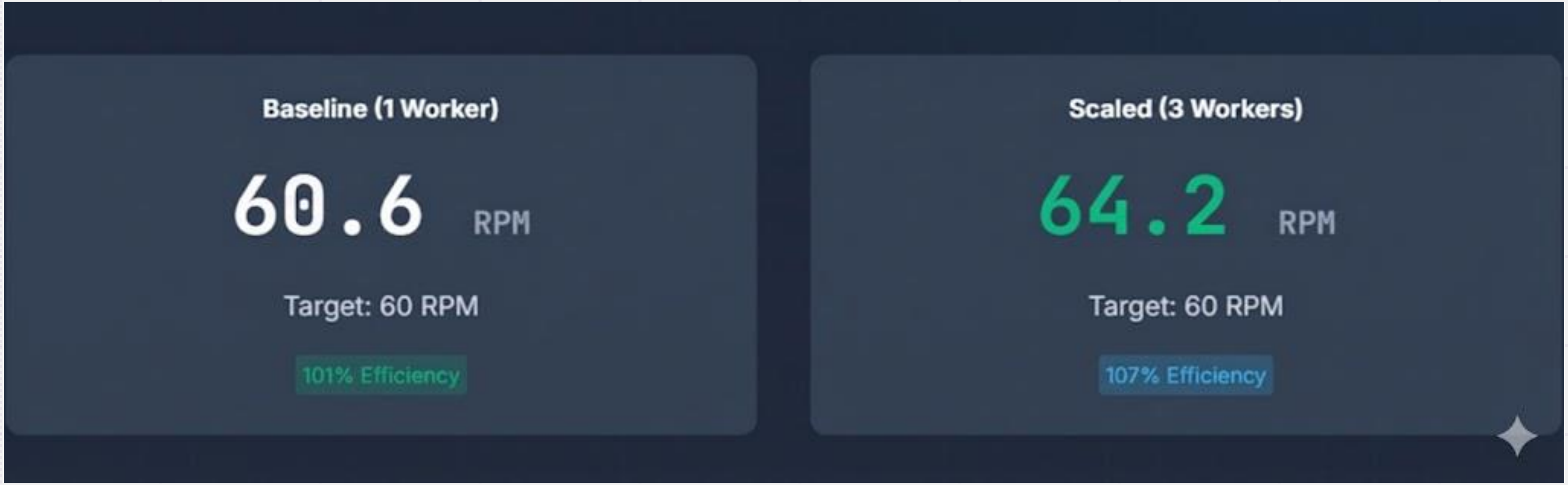
Evaluation II: Latency Analysis



Breakdown of end-to-end processing time per micro-batch.

- Despite Spark being the dominant factor, total latency is **1.3 seconds**, well within the sub-minute requirement.

Evaluation III: Scalability



Baseline (1 Worker)

60.6 RPM

Target: 60 RPM

101% Efficiency

Scaled (3 Workers)

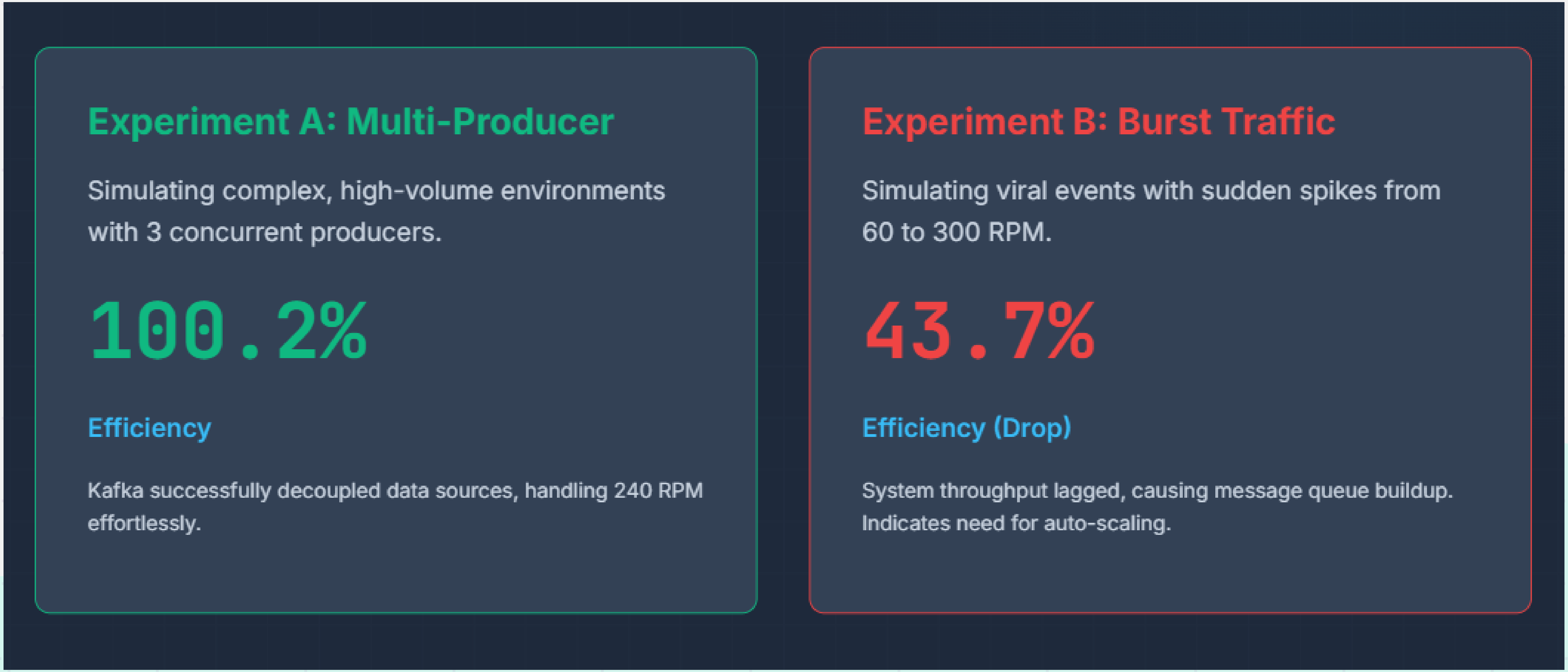
64.2 RPM

Target: 60 RPM

107% Efficiency

- **Resource Usage:** Memory usage scaled linearly (~3-9% increase per worker), confirming the system can safely accommodate additional nodes without resource exhaustion.

Evaluation IV: Robustness



Experiment A: Multi-Producer

Simulating complex, high-volume environments with 3 concurrent producers.

100.2%

Efficiency

Kafka successfully decoupled data sources, handling 240 RPM effortlessly.

Experiment B: Burst Traffic

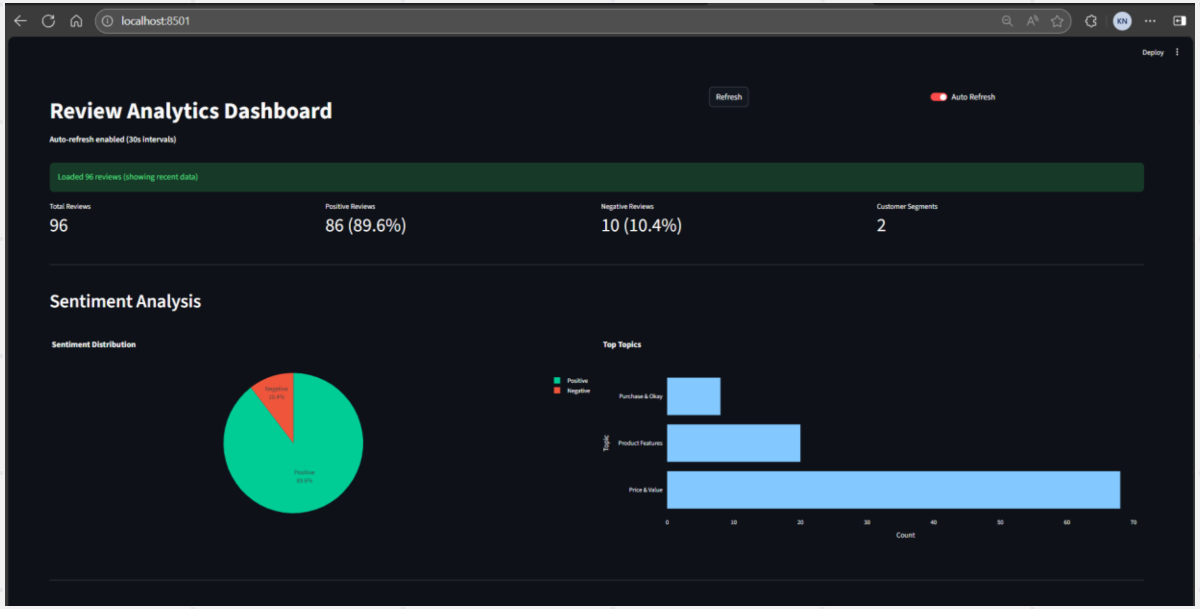
Simulating viral events with sudden spikes from 60 to 300 RPM.

43.7%

Efficiency (Drop)

System throughput lagged, causing message queue buildup. Indicates need for auto-scaling.

The Dashboard



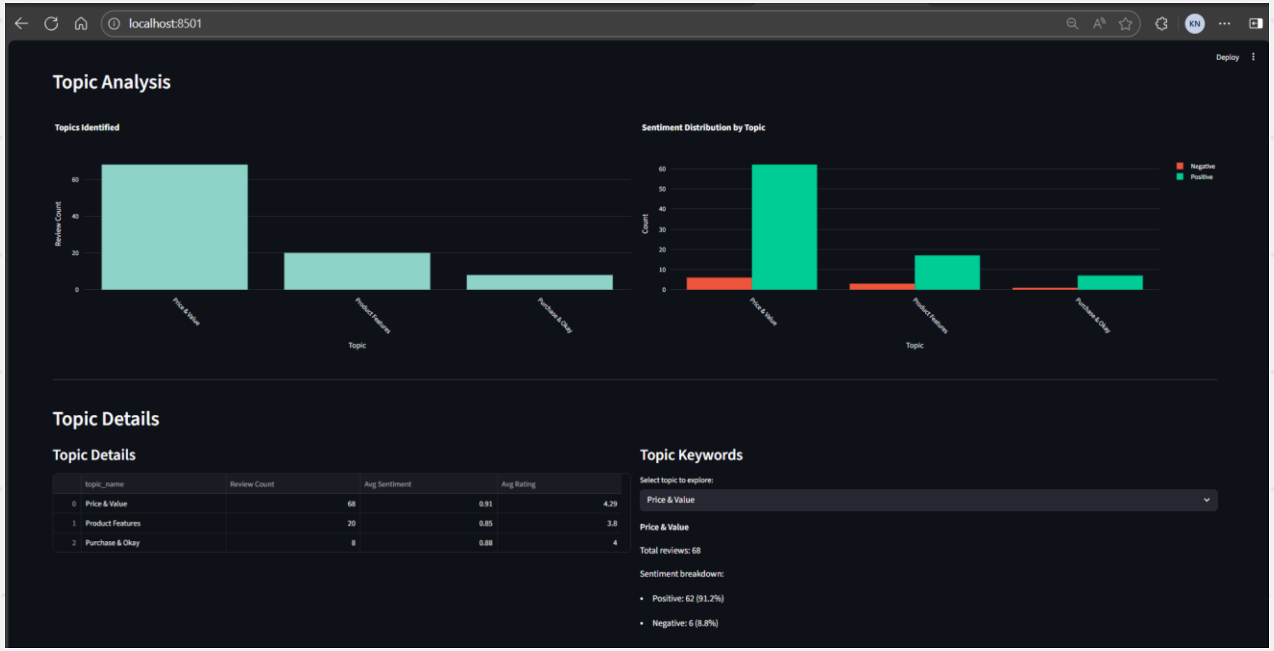
Review Analytics Dashboard

Auto-refresh enabled (30s intervals)

Loaded 96 reviews (showing recent data)

Total Reviews	Positive Reviews	Negative Reviews	Customer Segments
96	86 (89.6%)	10 (10.4%)	2

Sentiment Analysis



Topic Analysis

Topic Details

Topic Details

	topic_name	Review Count	Avg Sentiment	Avg Rating
0	Price & Value	68	0.91	4.29
1	Product Features	20	0.85	3.8
2	Purchase & Okay	8	0.88	4

Topic Keywords

Select topic to explore:

Price & Value

Price & Value

Total reviews: 68

Sentiment breakdown:

- Positive: 62 (91.2%)
- Negative: 6 (8.8%)

Limitations and Future Scope

- **Current Limitations**
  - **Single Machine Deployment**
  - **MongoDB Concurrency**

    Risk of write-contention performance drops if worker nodes scale out significantly.
  - **Batch-based Logic**

    Anomaly detection relies on stateless micro-batches; Model vocab is static/limited.
- **Future Scope**
  - **Kubernetes Orchestration**

    Deploy workers on K8s for dynamic auto-scaling and robust resource management.
  - **Stateful Stream Processing**

    Implement Window Aggregation for smoother anomaly detection and trend analysis.
  - **Enhanced ML Models**

    Expand vocabulary and explore Transformer-based embeddings for better topic interpretation.

Summary

Delivers a scalable system for real-time product issue detection and long-term trend analysis.

Uses Apache Kafka and Apache Spark to combine streaming sentiment analysis with batch-based topic modeling.

Employs MongoDB and Streamlit for efficient storage and intuitive visualization of insights.

Enables quick identification of rising negative sentiment, recurring defects, and other actionable patterns from unstructured review data.