setronica

CASE STUDY

Optimizing Real-Time Data Science Operations for a Leading Fashion and Apparel Company

Discover how Setronica's partnership with a top-tier Fashion and Apparel leader resulted in over 30x reduction in operational expenses, alongside dynamic real-time recommendations and trend analysis, revolutionizing the industry landscape.

THE CLIENT

Setronica collaborated with a dominant force in the Lifestyle > Fashion and Apparel sector, securing a spot among the top 100 industry players in the US. In an arena teeming with prominent brands like Shop Gap, Macy's, SHEIN USA, Nike, Nordstrom, and Everlane, our client stands out as a fierce competitor.

At the heart of our client's operations lies flash sales and e-commerce, where they wield a platform that beckons a staggering monthly user base of 4 million. What sets them apart is their distinct focus on exclusivity and strategic sales events, carving a unique niche in the cutthroat landscape of the industry.

OVER 30x OPERATIONAL EXPENSE REDUCTION

In a remarkable display of expertise, our team orchestrated a comprehensive four-phase AWS fine-tuning initiative.

The outcome was truly extraordinary: operational expenses plummeted from several thousand dollars per month to an astonishingly minimal $80, marking a reduction of over 30 times.

THE RESULT

We collaborate closely with a dedicated client-side data science team comprising six skilled individuals. Leveraging our specialized expertise, we assisted this data science team in crafting a dynamic real-time recommendation service.

This innovative service enables two key capabilities:

- **Personalized Real-Time Forecasting:** Our solution empowers the client to deliver tailored recommendations to customers in real-time, leading to increased sales by catering to individual preferences.
- **Dynamic Activity Trend Analysis:** By swiftly calculating activity trends for products and product collections, our service provides invaluable insights that aid in proactive decision-making and strategy refinement.

ENCHANCED STEPS FOR CHANGING THE FORECAST MODEL

1. Real-time Personalized Predictions

The service is fed by real-time data collected from websites and mobile applications.

Data events are processed through a pipeline of AWS Lambda functions and passed to the AWS Sagemaker real-time inference endpoint. The pipeline creates a rolling window for each website/application user to collect a batch of events used for inference. Inference results are stored in the DynamoDB table, and consumers can retrieve them through the API gateway. Health monitors, alerts, and dashboards have been created to simplify service support.

Our customer uses an ML model that is not supported by Sagemaker, so a custom Docker image was created to meet Sagemaker API requirements.

This implementation was created as an alternative to a solution based on an Apache Spark structured streaming POC project offered by the customer. The Lambdas-based implementation was accepted because it reduced the cost of the service from several thousand dollars to only $80 per month.

2. Dynamic Real-Time Activity Trend Computation

The service is fed by real-time data collected from websites and mobile applications.

Data events are processed through a pipeline of AWS Lambda functions with DynamoDB as the persistent data store. The DynamoDB CDC data stream was used for low-cost data processing. Rolling window function was used for data aggregation. Variable time width was implemented for the function to account for significant differences in day/night user activity. Results are available to consumers via API implemented using AWS API Gateway and Lambda function.

3. Apache Airflow Migration

Self-hosted Airflow has been migrated to the AWS MWAA service. DAGs code was migrated from the old Airflow to the latest version supported by MWAA.

Massive code refactoring was performed to make the code follow the best practices recommended by Airflow. This refactoring significantly reduced the load on the Airflow database and improved the stability of the MWAA service.

4. Ingenious Databricks Jobs Optimization

The data science team uses Spark jobs on the Databricks platform to perform model training and inference. These jobs represent a significant portion of the team's budget.

Some of the most expensive jobs were optimized to reduce their cost. By splitting the jobs into parallel tasks, the cluster resources were used more efficiently and costs were reduced by 30%.

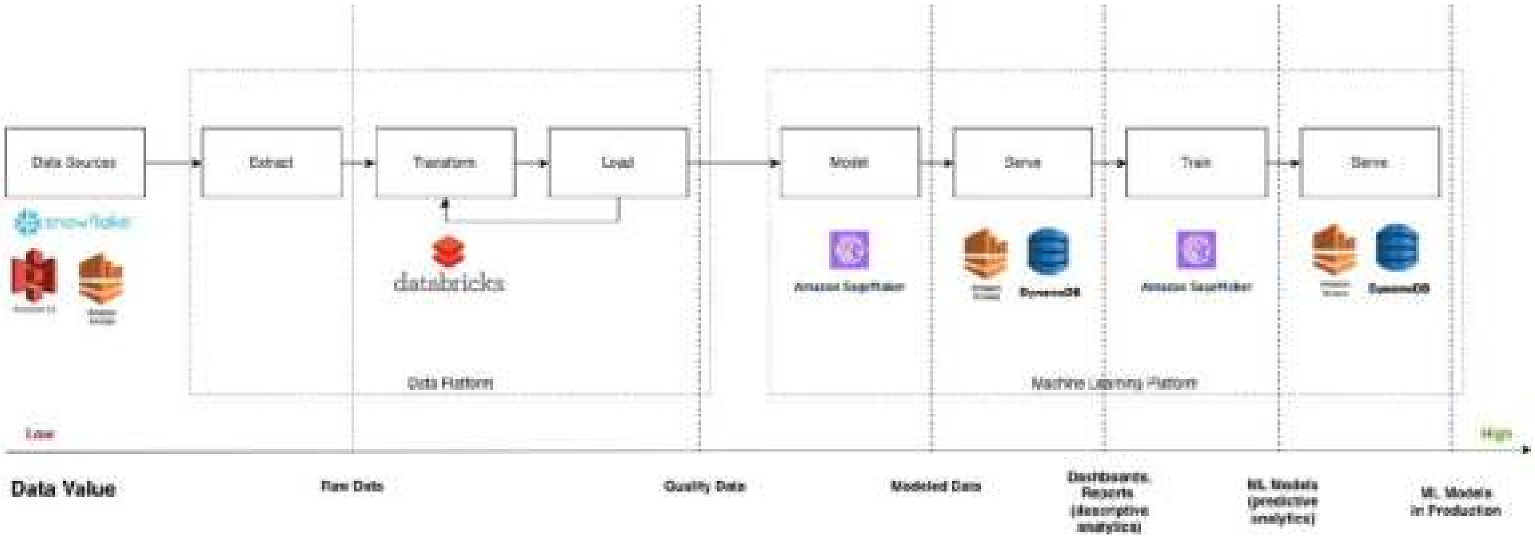
Another example — the data processing graph was restructured and the cost was reduced by 3 times.

THE TEAM

Senior Java developer and 6 specialists from the client-side data science team.

DATA VALUE CHAIN

The Data Value Chain Model encapsulates data's journey from raw material to actionable insights. It encompasses stages like collection, processing, analysis, and interpretation, yielding valuable outcomes. This case study exemplified this model in transforming services and optimizing processes, highlighting our dedication to extracting optimal value from data for business success.



ENCHANCED TECHNICAL STACK

AWS Databricks	Analytical service for processing big data (compatible with Apache Spark code).	Facilitating seamless aggregation and processing of large-scale data.
Snowflake	Big data dataset storage.	Providing a robust repository for storing and managing datasets.
AWS Sagemaker	ML-model training and deployment.	Empowering the creation, training, and deployment of accurate machine learning models for predictive inferences.
AWS S3	Scalable object storage system.	Offering a reliable and versatile solution for storing and managing datasets.
AWS Kinesis	Real-time data streaming service.	Enabling efficient collection, processing, and analysis of real-time data streams with scalability.

THE CI/CD WORKFLOW

1. We split the code from the mono-repository into separate repos and created a set of rules for creating new repos and services – so there is repeatability and quality.
2. Once a developer gets a task – he creates a branch in a given repo to isolate his code.
3. Then, after making pull-request, we run tests and deploy to test environment.
4. If tests ran well – developer makes pull-request to merge code into master branch.
5. We set up autodeployment to run immediately after pull-request is approved. So the code on the production environment is the same as on the master branch.

Setronica's strategic collaboration with this Fashion and Apparel industry titan underscores our unwavering commitment to excellence and innovation.

Through data science acumen and intricate technical implementations, we have catalyzed profound enhancements in real-time recommendations, trend calculations, and operational efficiency. The reduced costs, streamlined workflows, and client satisfaction stand as a testament to our expertise in elevating data-driven operations.

LET'S START BUILDING SOMETHING GREAT TOGETHER!

setronica.com

contact@setronica.com

SLOVENIA Kolodvorska 7, 1000 Ljubljana
USA 211 E 7th St, Austin, TX 78701