BIG DATA PROJECT

GONNET Mathéo
GOULAMHOUSSEN Ines

TABLE OF CONTENTS

Who are we?

This report is produced for our client ShopSphere, a company specializing in **e-commerce**. ShopSphere is all about **customer satisfaction**, whether it's selling the latest gadgets or everyday essentials. But with thousands of customers leaving **reviews**, posting on **social media**, and **rating products** daily, the team is feeling a little overwhelmed. **How can this mountain of data be turned into strategic gold?**

That's where we, consultants from **Adaltas**, step in!
We're a dynamic duo of experts who enjoy tackling technological challenges as much as debating who ate the last cookie in the meeting room. **Joe**, our Spark maestro, handles everything related to the fast processing of massive datasets. **Yanis**, the Kafka specialist, skillfully orchestrates the seamless flow of real-time data streams. Together, we've been tasked with designing a **distributed processing platform** to help ShopSphere harness this data and extract valuable insights.

With a blend of cutting-edge technologies, a dash of scalability, and a sprinkle of real-time visualization, we're ready to turn a complex challenge into a sleek and powerful solution.

Yanis

Joe

Project Objectives

The main goal of this project is to enable **ShopSphere** to better understand and leverage customer feedback to improve its strategy and operations. Through sentiment analysis, ShopSphere will be able to:

- **Classify customer reviews** into positive, negative, or neutral sentiments to identify strengths and areas for improvement.
- **Detect trends** in customer feedback over time and across product categories, helping anticipate expectations and adjust offerings.
- **Provide real-time insights** through intuitive dashboards, allowing teams to make informed decisions quickly.
- **Integrate multi-source data**, including inputs from the e-commerce site, social media platforms, and review sites, for a comprehensive and consistent understanding of customer expectations.

By creating a robust and scalable platform, this project aims to transform raw data into actionable strategic insights, ensuring ShopSphere's sustainability and competitiveness in the e-commerce market.

Global Architecture

The system architecture for ShopSphere's sentiment analysis is based on a three-layer structure, where each layer has its own job to do—like a team working together to get the best results.

1. Overview of the Architecture

The architecture is divided into **three** main layers:

- **Data Ingestion Layer**: Collects real-time data from different sources and sends it down the pipeline.
- **Data Processing Layer**: Cleans the data and analyzes it—kind of like polishing raw diamonds into something valuable.
- **Storage and Visualization Layer**: Keeps the processed data safe and presents it with clear and interactive dashboards.

2. Description of the Three Main Layers

A. Data Ingestion Layer (Apache Kafka)

In this layer, **Apache Kafka** plays the role of a **data traffic controller**, managing the constant flow of information from different sources like customer reviews, social media posts, and product ratings. Think of it as a **giant sorting center for data**, where packages (data) arrive from various producers, such as **APIs** or **web forms**. Kafka carefully organizes this data into **"topics"** (like putting packages into labeled bins) to keep everything neat and ready for processing.

But Kafka isn't just about sorting; it also ensures that nothing gets lost along the way. All data is stored **safely and reliably**, even if something goes wrong in the system. This makes it the ultimate **middleman** for handling **real-time data** before it moves on to the next stage of our system.

In short, Kafka is like the **train station** of our architecture, where every piece of data knows exactly which track it needs to take.

B. Data Processing Layer (Apache Spark)

In this layer, **Apache Spark** takes on the role of the **data workshop**, where raw data is cleaned, polished, and transformed into valuable insights. Spark pulls data directly from Kafka using **Spark Streaming**, making them a bit like **best friends who work together seamlessly**.

Once the data arrives, Spark gets to work by **cleaning it up** (removing typos, unnecessary characters, and anything that might cause confusion). It then **breaks the data into smaller, manageable pieces** and uses its tools to analyze customer sentiments, categorizing them as positive, negative, or neutral. This step is like running messy reviews through a **translation machine** that turns chaos into clarity.

When Spark finishes its job, the results are passed along to the next stage.
Continuing with our train metaphor, Spark is the **repair shop** at the station where every data package is inspected, fixed, and stamped with a clear label before it heads out on the next train.

C. Storage and Visualization Layer (Elasticsearch & Kibana)

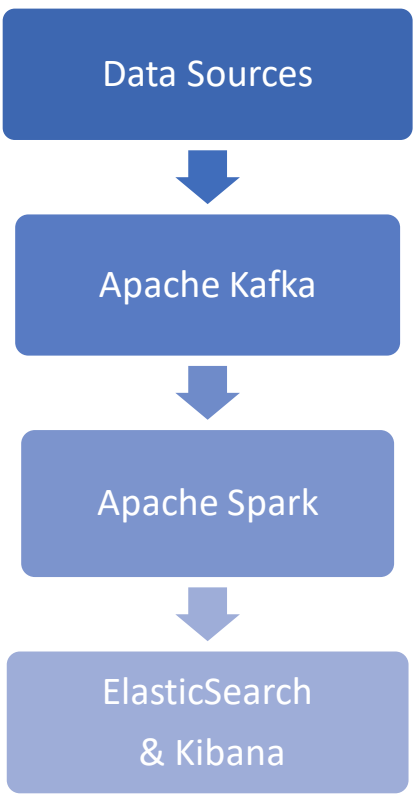
This is the final stop for the data, where **Elasticsearch and Kibana** work together to store the processed information and turn it into **easy-to-read dashboards**. **Elasticsearch** acts like a **super organized library**, where every piece of data is indexed and stored, so it can be quickly searched and analyzed whenever needed.

After that, **Kibana** takes over and creates **visual dashboards** that show trends, insights, and key numbers in a way that's simple to understand. These dashboards help teams like **marketing** and **customer service** see what's happening and make decisions more easily.

If we go back to our train metaphor, Elasticsearch is the **arrival platform**, organizing all the packages (data) that Spark has prepared. Kibana is the **information board**, displaying the data clearly for everyone to use. Together, they make sure that all the hard work done along the journey ends up being useful and easy to understand.

3. Data Flow Diagram

The diagram below shows how the data flows step by step:



Ingestion: Collects raw data and organizes it

Processing: Spark cleans and analyzes the data to understand customer sentiments

Storage: Stores the processed data for quick access
Visualization: Displays everything on dashboards that help teams take action.

Technologies used

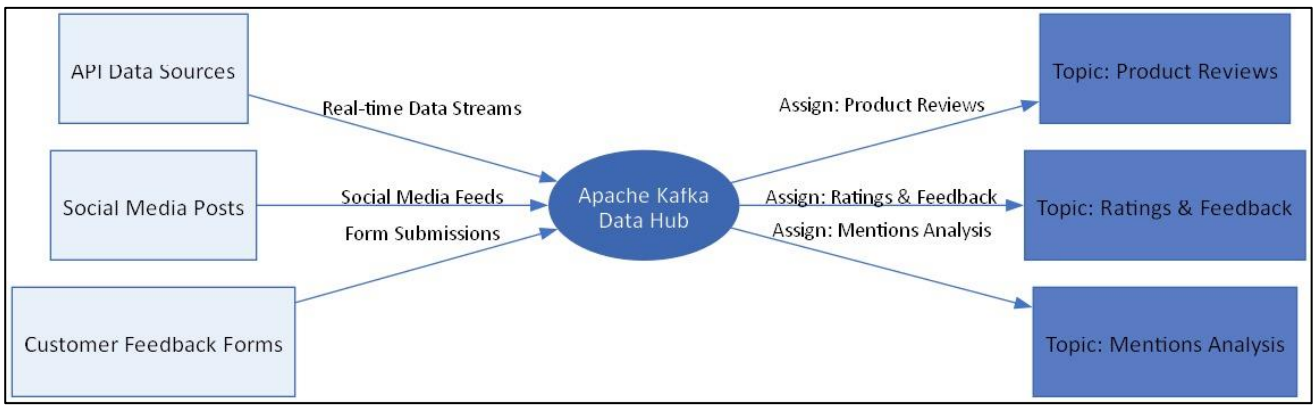
Now that we've decided on the technologies for ShopSphere's sentiment analysis project, let's **dive deeper** into how each will function within our mission. We'll explore their roles, using specific examples to connect the tools to our use case.

Apache Kafka: The Real-Time Traffic Controller

In our project, Apache Kafka acts as the central hub for **all incoming data**. Imagine a busy train station, where **reviews**, **social media posts**, and **product ratings** arrive from various sources like customer feedback forms and API streams. Kafka ensures every piece of data is **assigned to the right "track**" (a topic like "Headphones Reviews" or "Social Media Mentions").

> For example, if a customer tweets, *"Love the sound quality of my new headphones, but the ear pads are uncomfortable, "*Kafka immediately captures this tweet, assigns it to a relevant topic (like "Product Reviews - Headphones"), and ensures it's ready for processing. Even if there's a glitch in the system (e.g., Spark is momentarily down), Kafka safely holds the data until everything is back on track.

In short, Kafka ensures that all customer feedback, whether structured (star ratings) or unstructured (social media posts), **flows seamlessly and reliably into the next stage of our pipeline.**



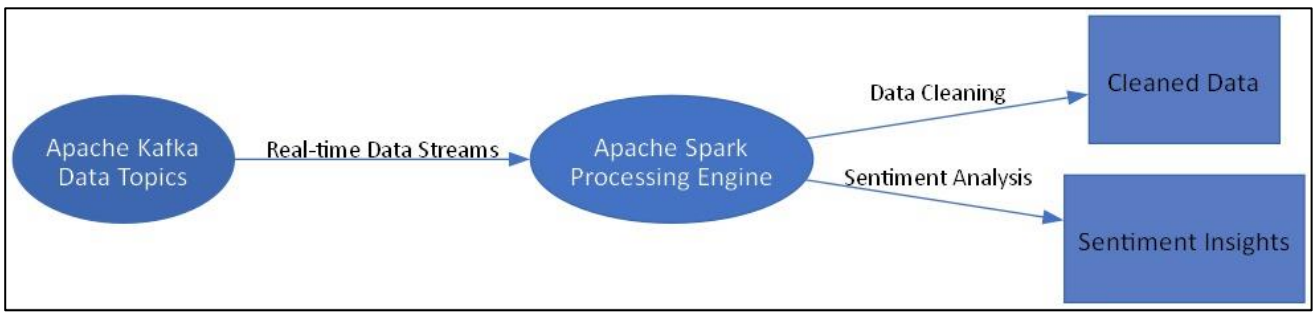
Kafka Data Pipeline for Customer Feedback Analysis

Apache Spark: The Data Workshop

Apache Spark is where the raw data sent by Kafka is **transformed into actionable insights.** Think of it as a high-tech workshop where every piece of feedback is cleaned, analyzed, and categorized.

For instance, let's say Kafka sends a flood of social media posts about a product launch. Spark's job is to first **clean** this data, removing typos, emojis, and irrelevant content like hashtags (#bestproductever). Next, it applies Natural Language Processing (NLP) techniques to figure out the sentiment behind each post.

> Take the example, *"Amazing delivery speed, but the packaging was damaged!"* Spark breaks this down into two sentiments: *positive* for delivery speed and *negative* for packaging. It then tags this feedback appropriately and calculates aggregate trends, such as *"80% of reviews mention fast delivery positively."* These insights are passed to Elasticsearch for indexing.



Spark Data Pipeline for Real-Time Sentiment Analysis

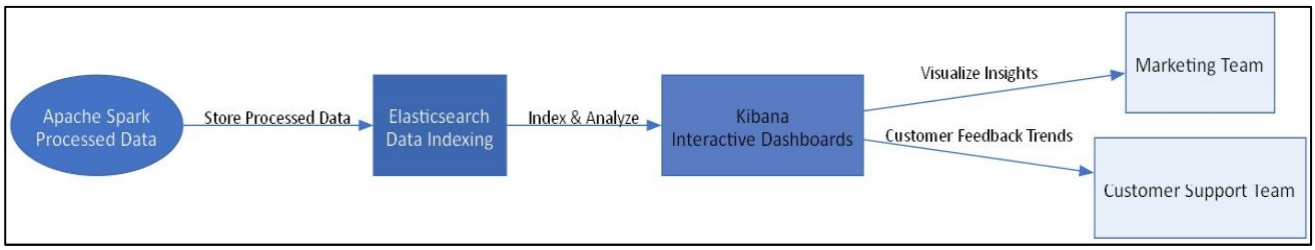
E

ElasticSearch & Kibana: The Insights Builders

Once Spark has done its job, the processed data is **stored** and **visualized** using Elasticsearch and Kibana. Elasticsearch is like a super-efficient librarian who makes it easy to search for any piece of data. Kibana, on the other hand, turns this information into clear, visual stories.

> For example, suppose ShopSphere's marketing team wants to know, *"How do customers feel about our headphone lineup this month?"* Elasticsearch allows them to **quickly** filter through all reviews tagged with "Headphones" and "January 2024." Kibana then creates an interactive **dashboard** showing key trends: *"Positive sentiment is up by 20%, but mentions of ear pad discomfort have increased by 15%."*

These dashboards empower teams across ShopSphere—whether it's marketing adjusting ad campaigns or product development tweaking designs—to make informed decisions based on real-time data.



Data Processing and Visualizations Pipeline with Apache Spark and Elastic Search

Complementary tools: The Behind-the-Scenes Heroes

While Kafka, Spark, and Elasticsearch form the backbone of our system, additional tools ensure seamless operation.

Docker creates a consistent environment for each component, eliminating the common "it worked on my machine" problem.
Kubernetes handles scaling effortlessly, ensuring the system can manage spikes in data traffic, such as the influx of reviews during Black Friday.

For text analysis, Spark's built-in capabilities process customer feedback effectively, categorizing phrases like "mind-blowing" or "not worth the hype" into appropriate sentiment labels.

Justification of Technology Choices

We've chosen technologies that best fit the needs of ShopSphere's sentiment analysis platform, focusing on scalability, speed, and reliability. Below is a simplified comparison showing why each was selected over alternatives.

Apache Kafka

Technology	Why Chosen	Why Not Others
Apache Kafka	Handles high-throughput, real-time data streams	RabbitMQ lacks scalability for streaming.
RabbitMQ	Good for simple messaging tasks	Not suitable for real-time data pipelines.
Amazon Kinesis	Managed and easy to set up	Vendor lock-in and less flexible.

Kafka ensures **reliable**, real-time handling of reviews and social media posts, outperforming RabbitMQ and Kinesis in scalability and flexibility.

Apache Spark

Technology	Why Chosen	Why Not Others
Apache Spark	Fast in-memory processing; handles NLP tasks well	Hadoop is too slow.
Hadoop MapReduce	Reliable for batch tasks	Inefficient for real-time processing.

Spark's speed and versatility make it perfect for sentiment analysis, **handling real-time streams** and NLP tasks with ease.

ElasticSearch & Kibana

Technology	Why Chosen	Why Not Others
Elasticsearch	Fast text search; scalable; integrates with Kibana	PostgreSQL struggles with unstructured text and is slower to scale.
PostgreSQL + Tableau	Good for structured data, but struggles with unstructured text	Slower and costly to scale for this use case.

Elasticsearch is ideal for **storing** and **searching** sentiment data, while Kibana provides intuitive **dashboards** for visualizing trends.

Conclusion

In conclusion, the solution we've designed for **ShopSphere** brings together the best of real-time data streaming, fast processing, and powerful visualization to transform customer feedback into actionable insights. By leveraging **Apache Kafka**, **Apache Spark**, and **Elasticsearch & Kibana**, we've built a robust, scalable system capable of handling the flood of reviews, social media posts, and product ratings that come with operating in the fast-paced e-commerce world.

This solution empowers teams across the company, from marketing to customer support, to make smarter, faster decisions, ensuring that ShopSphere stays ahead of the competition. And, most importantly, it's all done with a bit of flair, making the complex world of big data not just manageable but also a bit fun.

At this stage, we might want to start considering becoming teachers at an engineering school like ECE, alongside our consulting activities at Adaltas..

Key Sources

Apache Kafka Documentation
Apache Spark Documentation
Elasticsearch Documentation
Developpez.com - Forum Docker
OpenClassrooms - Gérez des flux de données en temps réel avec Kafka