

How a CRM company turned its customers' data into a Resilient Product Advantage in No-Time with Flexor

Transforming feedback into a valuable data product with an end-to-end LLM-powered text classification pipeline

Stack

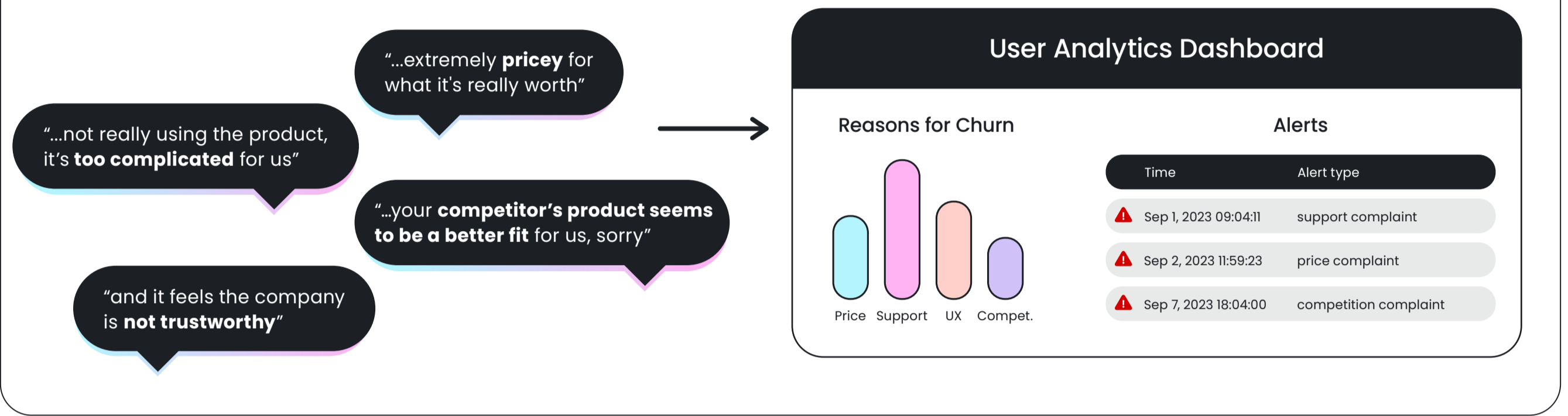
- **Text becomes a powerful product feature** – a well-known CRM company set out to capitalize on all its textual feedback data by building premium product features out of it.
- **Skipping the pain of building robust LLM pipelines** – with Flexor, the overwhelming amount of work required to set up their own infra & pipelines was replaced by a simple SQL query.
- **Production-ready in a matter of days** – a scalable & flexible text analysis pipeline brought to production by a single BI developer in less than 2 weeks.

[Book a demo here >](#)

The CRM Company's Challenge Transforming Feedback into Insights

The Head of Data at a well-known CRM company was seeking to enhance its customer analytics dashboard to improve customer retention and to outpace competition. She decided to furnish her customers with insights into why their end-users churn, relying on the end-users' feedback collected by the CRM platform.

What the Head of Data had in mind



The plan seemed simple:

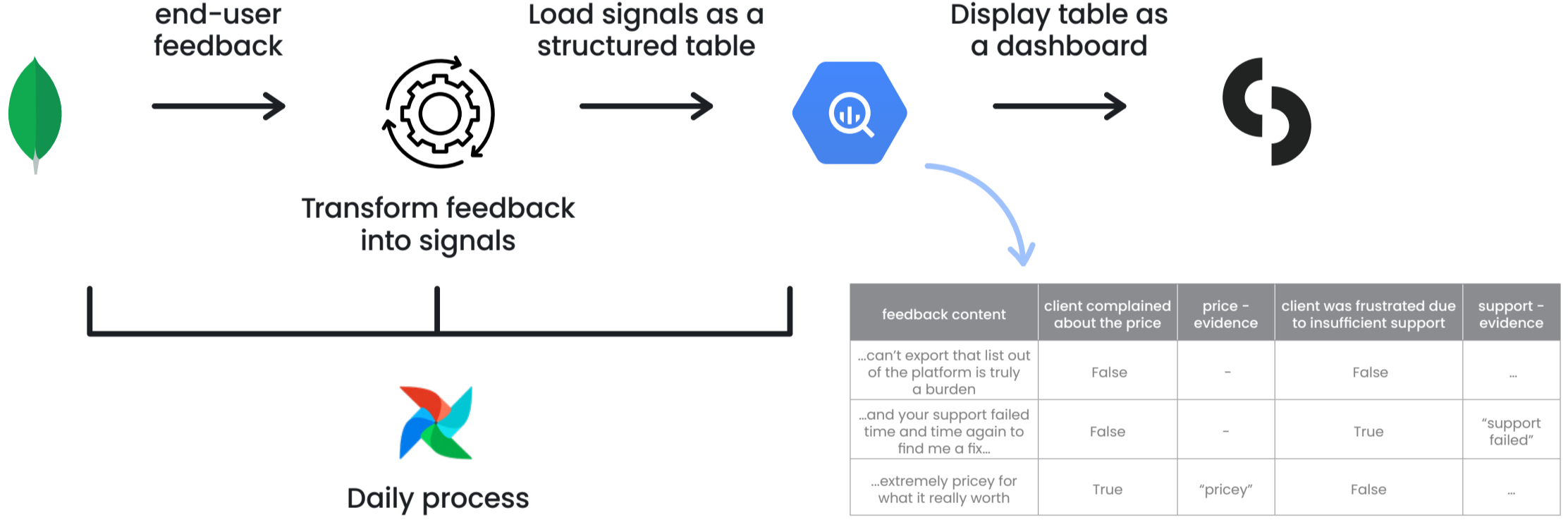
build a "user-insights data product" that enriches every feedback with signals inferred from the text, and make it easy for BI developers to transform it into insights, graphs and alerts with BigQuery and Sisense.

The table behind the dashboard

feedback content	client complained about the price	price - evidence	client was frustrated due to insufficient support	support - evidence
...can't export that list out of the platform is truly a burden	False	-	False	-
...and your support failed time and time again to find me a fix...	False	-	True	"support failed"
...extremely pricey for what it really worth	True	"pricey"	False	-

Building a robust text classification pipeline is complex

The chosen solution approach was building a daily text classification pipeline that extracts feedback entries from the CRM database, leverages NLP techniques to transform them into meaningful signals, and returns a well-structured tabular format as output. Those tables were to be uploaded into BigQuery that would serve as the data source of the analytics dashboard, via Sisense as an embedded dashboards engine.



Encountering hurdles

Problems to solve

With today's technology, this sounds pretty much straight forward, but in order to deliver significant value to the end-users and return positive ROI, **stringent requirements for both quality and scale** were essential:

- Signals are expected to be contextual and nuance-sensitive.
- Outputs must be structured to integrate well into existing data & analytics workflows, which are all based on structured data.
- The system has to produce consistent, high-quality outcomes, while being flexible and quick to support their fast product development cycle.

As a result, things became much more complicated for the data team



The Solution

Building an LLM-powered text classification pipeline with Flexor



Flexor's Unstructured Data ETL solved their problem end-to-end, without compromising on any of their solution requirements, with just two connectors and a few natural-language statements. All this, while fully complying with their stringent requirements:

E

Flexor's **source connectors** periodically **extract** feedback periodically from the company's MongoDB, and **prepare it for processing** by LLMs.

T

Flexor's **language processing engine** transforms the **text** and infers signals out of **each feedback**, according to **natural language instructions**.

L

The **destination connector** seamlessly loads the pipeline's output as a **structured table** into BigQuery – so they can easily analyze it in Sisense.

The Result

an entire LLM-based data pipeline delivered to production in days, by a single BI Developer

[Book a demo here >](#)

The CRM company successfully delivered one of its most engaging premium features while beating its development schedule by more than 6 months.