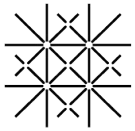
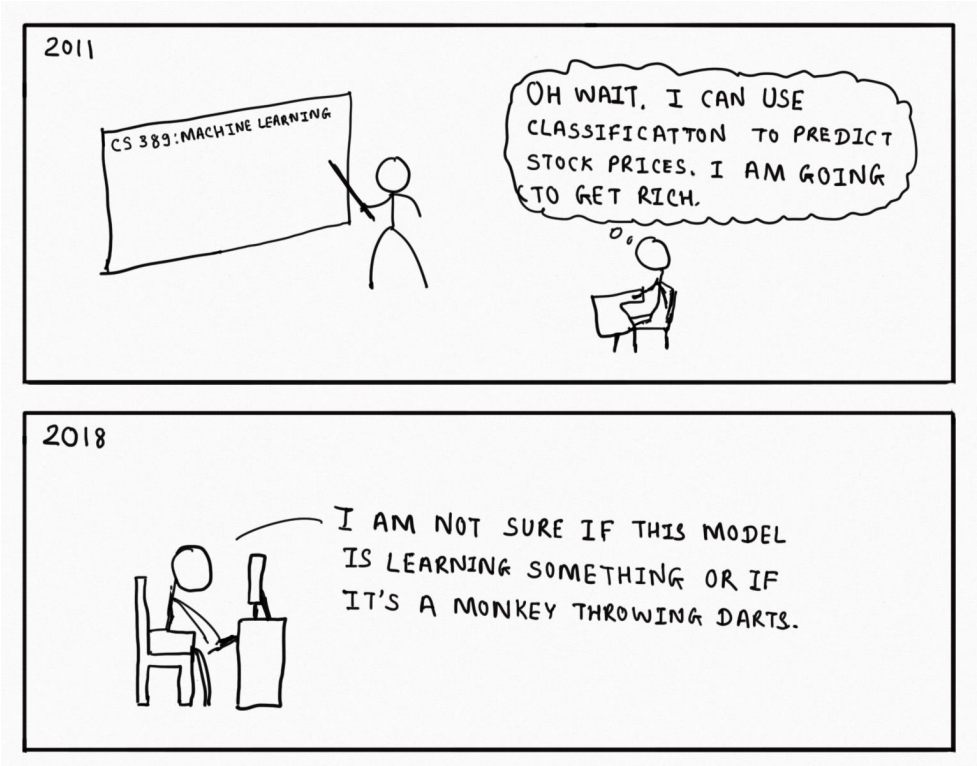


University of Basel

Workshop Report:

Stock Price Prediction System using Finnhub Data-stream and Apache Spark

Anastasiya Merkushova, Raphael Waltenspül

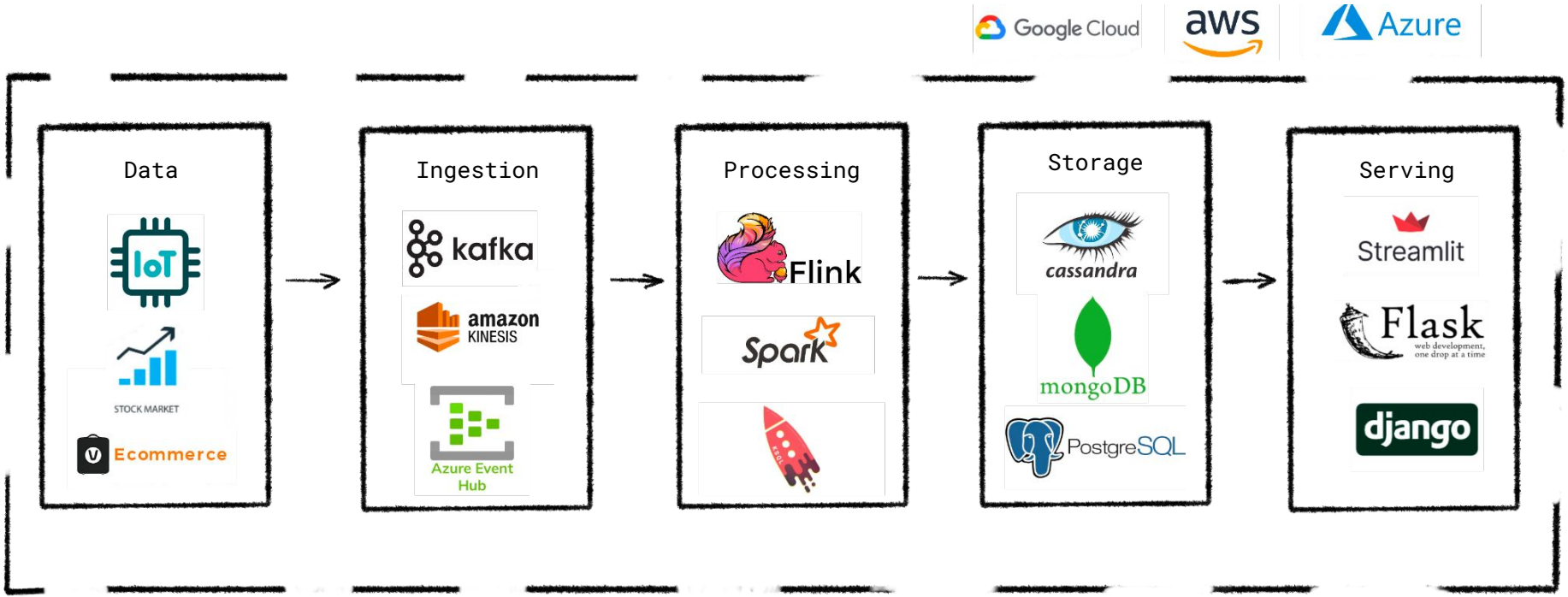

2011
CS 389: MACHINE LEARNING
OH WAIT. I CAN USE CLASSIFICATION TO PREDICT STOCK PRICES. I AM GOING TO GET RICH.
2018
I AM NOT SURE IF THIS MODEL IS LEARNING SOMETHING OR IF IT'S A MONKEY THROWING DARTS.

Goals and Objectives

- **Goal:**
  - Build a real-time distributed system to predict stocks price

- **Objectives:**
  - Design a pipeline for real-time data processing and price prediction
  - Identify and choose the optimal services, tools, and libraries
  - Develop and integrate all components of the system
  - Create a user-friendly interface to display data

Architecture

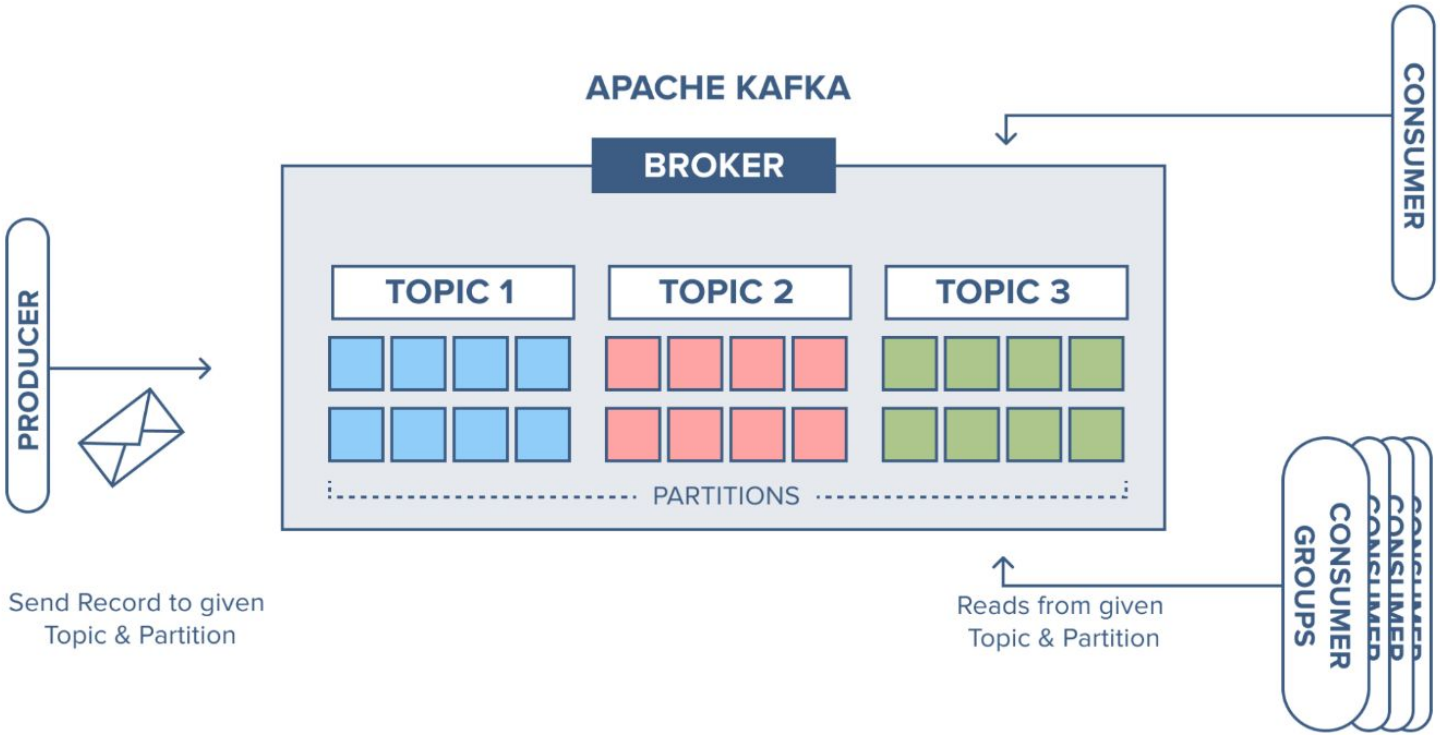


Data streaming pipeline

- **Why Finnhub?**
  - Data streaming via WebSockets, instant updates for stock prices
  - Offers stocks, forex, and cryptocurrencies, covering global markets
  - Free, detailed documentation, available for different languages

- Why Kafka?
  - Handles large volumes of data streams
  - Acilitates real-time data processing, enabling quick reaction to events
  - Data availability by replicating data across multiple brokers
  - Decouples producers and consumers, allowing asynchronous communication

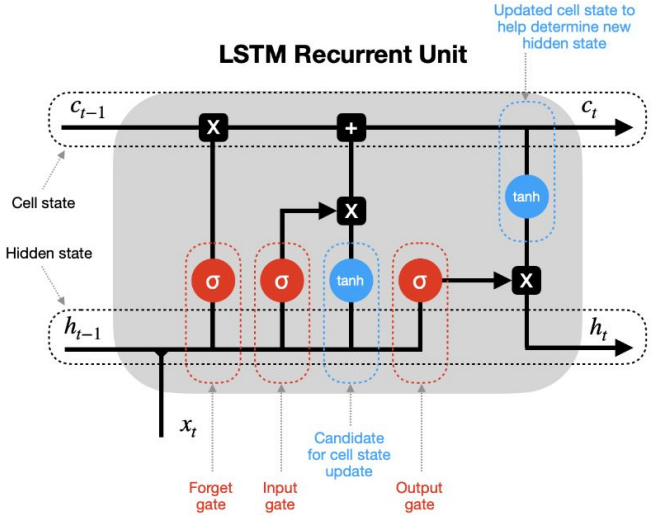
Ingestion: Kafka



[*] https://kafka.apache.org/

- **Why Apache Hadoop?**
  - Distributed file system (HDFS) and processing framework (MapReduce) provide built-in fault tolerance
  - Excels in processing vast amounts of data across distributed clusters
  - Supports various models, including batch processing, interactive querying, and real-time processing

Processing: Model Training



with TensorFlow

- LSTM able to selectively forget information over time
- TensorFlow optimizes computation and allows easy experimentation with different model architectures and hyperparameters

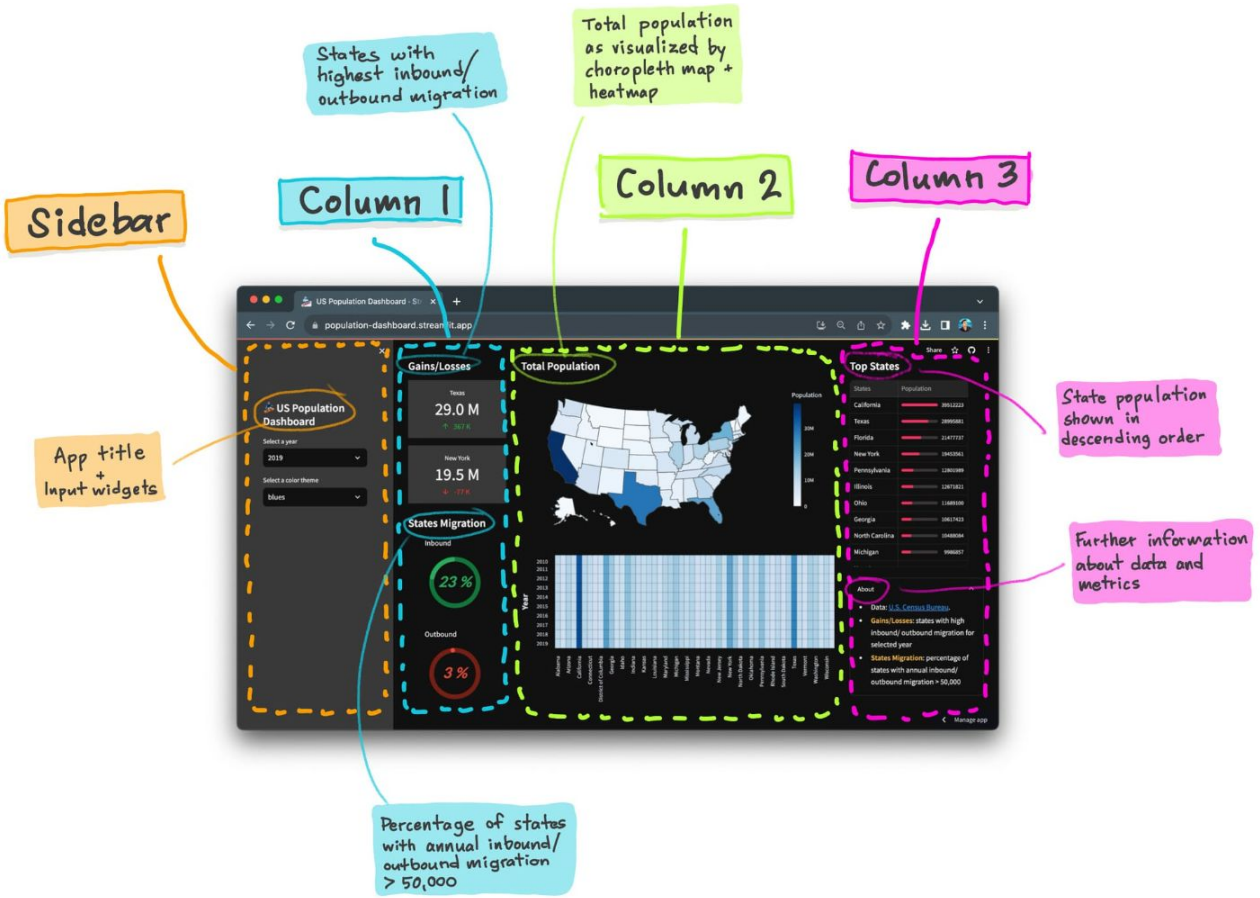
[*] towardsdatascience.com

- **Why MongoDB?**
  - Non-relational and document-oriented database
  - High performance for read and write operations
  - Horizontally scalable
  - Support for real-time data processing and aggregation, well integrated with analytical frameworks

Streamlit

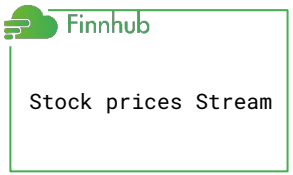
- **Why Streamlit?**
  - Python-based framework, no need for JS or CSS
  - Declarative syntax and automatic layout management
  - Providing built-in support for popular data science libraries

Visualization: Streamlit



Implementation


Finnhub
Stock prices Stream

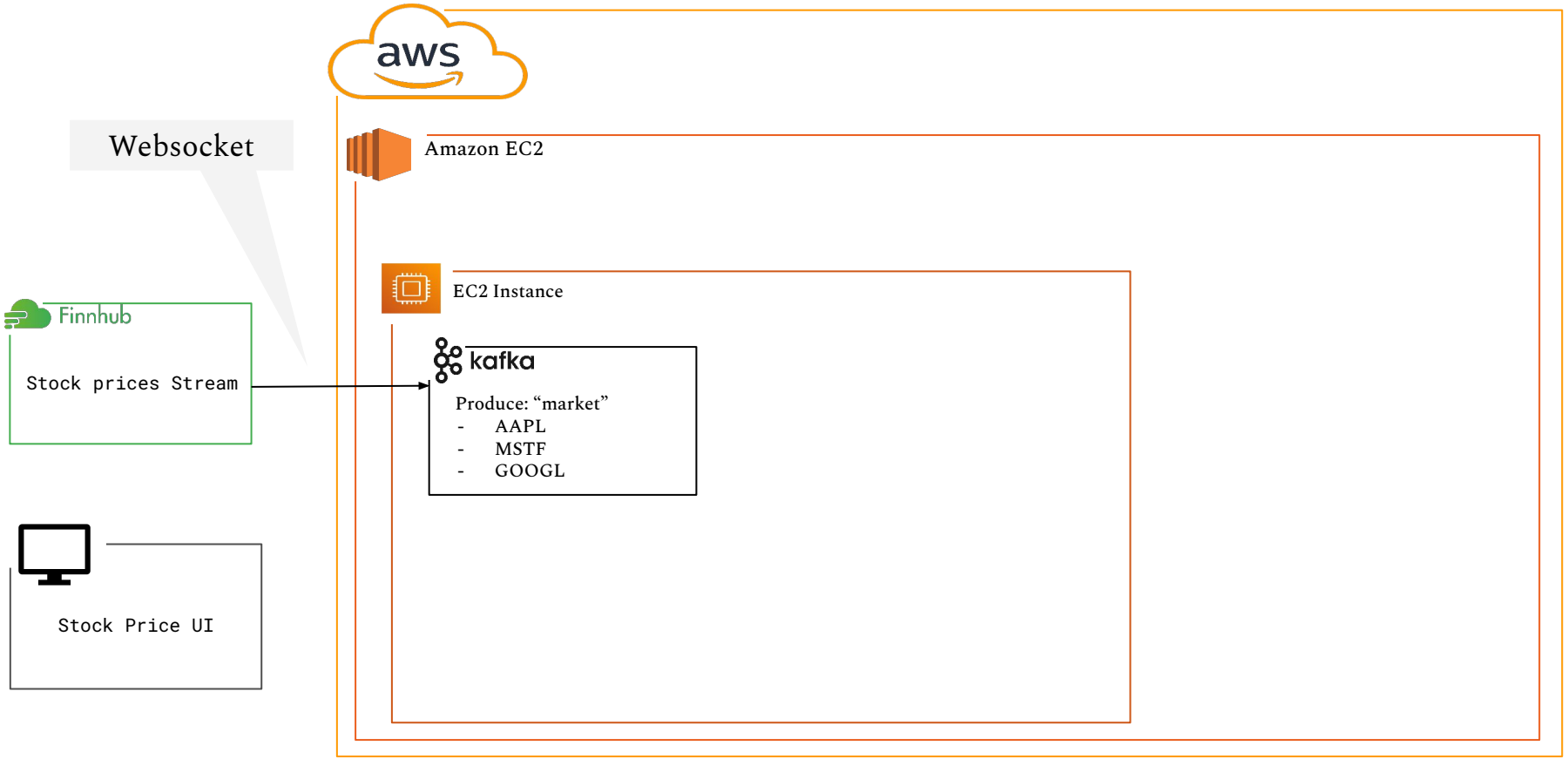

Finnhub
Stock prices Stream


Stock Price UI

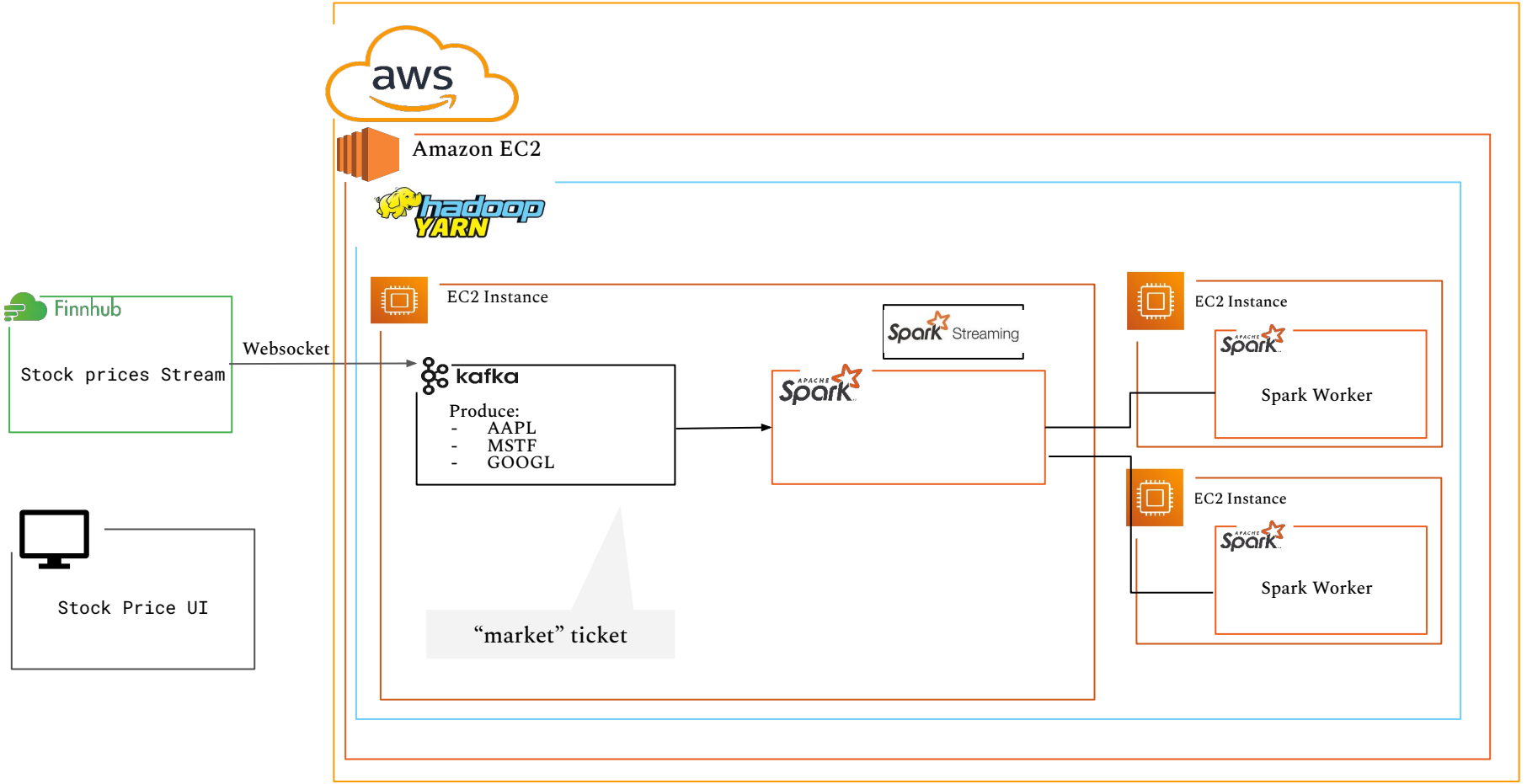
Implementation

Implementation

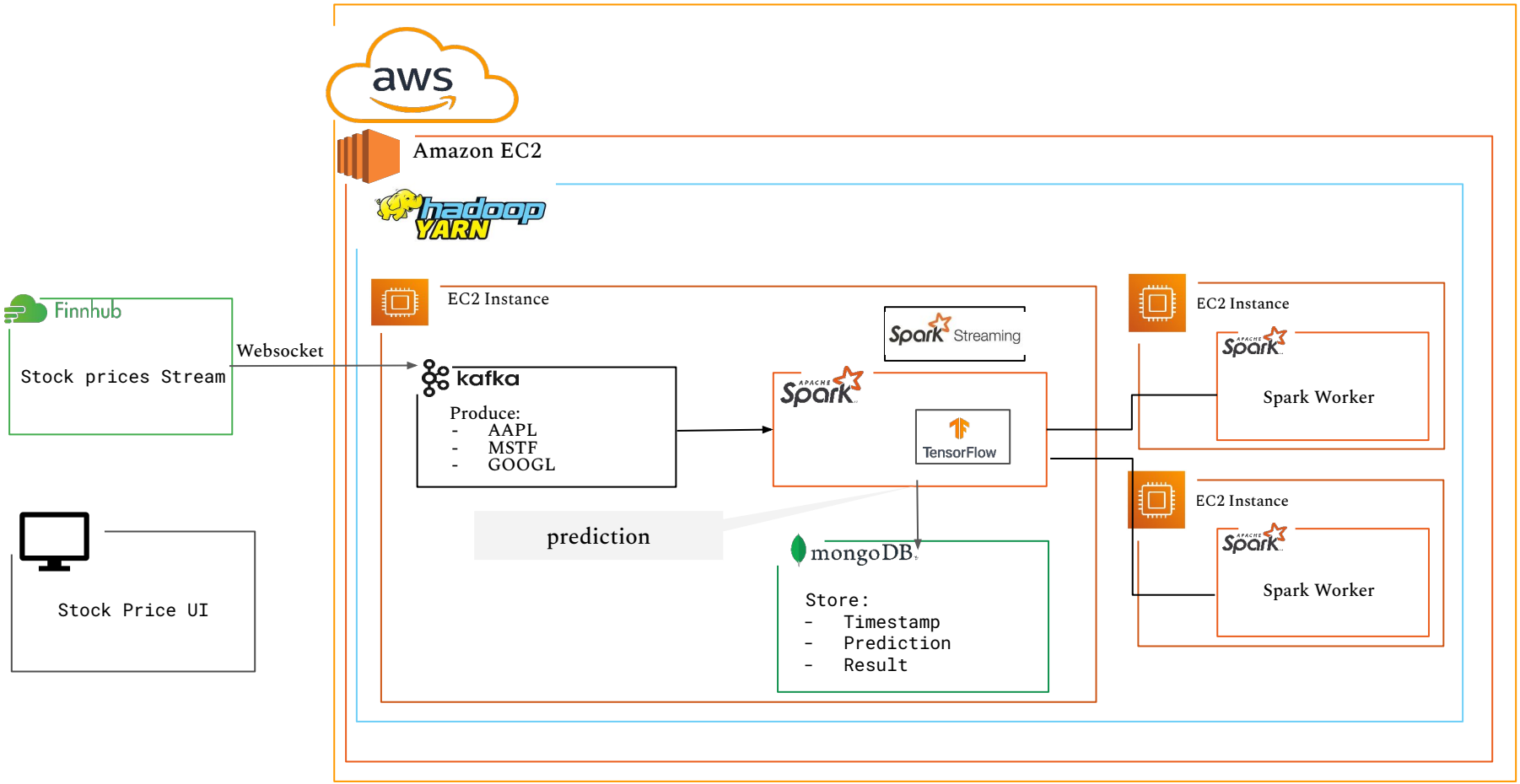
Implementation



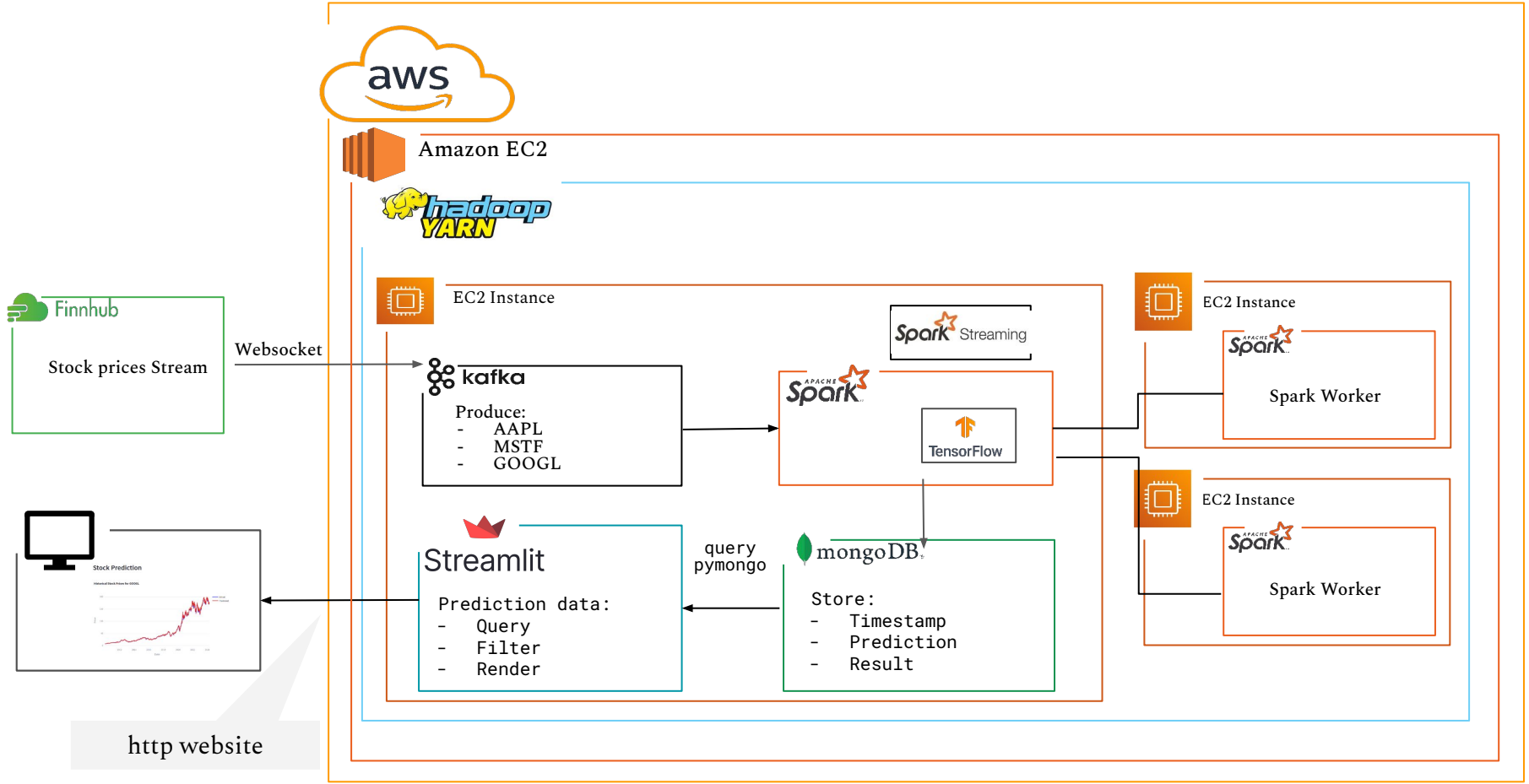
Implementation



Implementation



Implementation



Demo

Results:

- We built a scalable distributed system for predicting stocks

Conclusion

Results:

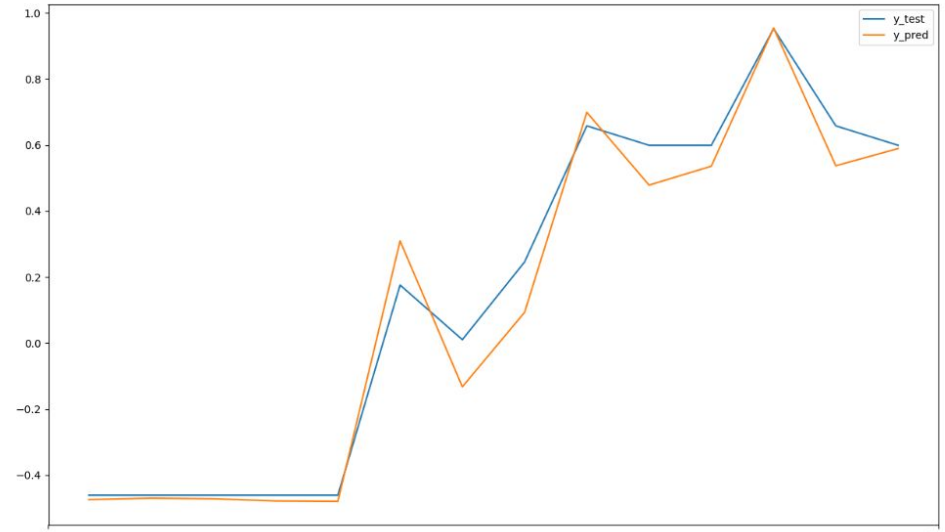
- We built a scalable distributed system for predicting stocks

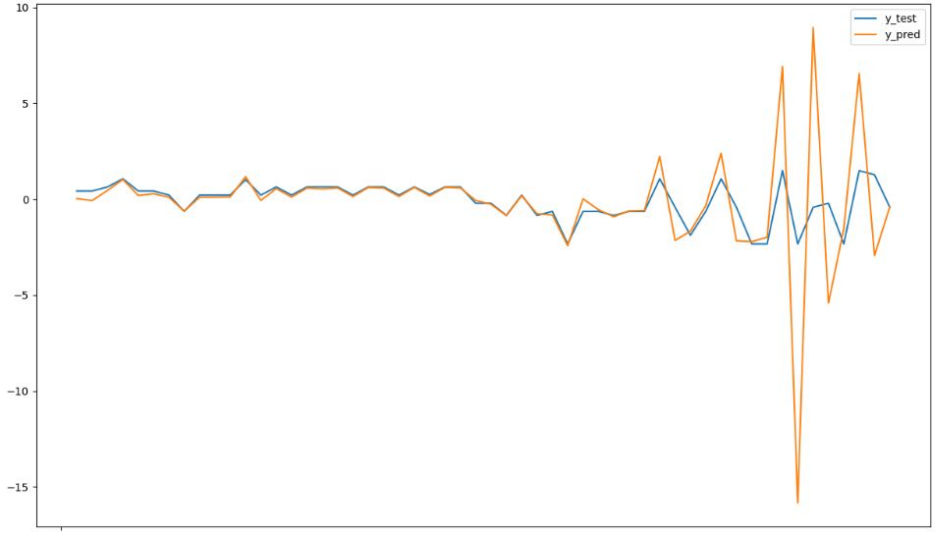
Future work:

- Use Spark MLlib for distributed model training to improve scalability
- Containerize each part using Docker to ensure reproducibility
- Integrate CLIP or a similar model for real-time feature extraction on streaming data

Thanks for the attention!

Questions time!


y_test
y_pred
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4


y_test
y_pred
10
5
0
−5
−10
−15

Learnings:

- About RTFM
- About documentation.
- About Complexity.