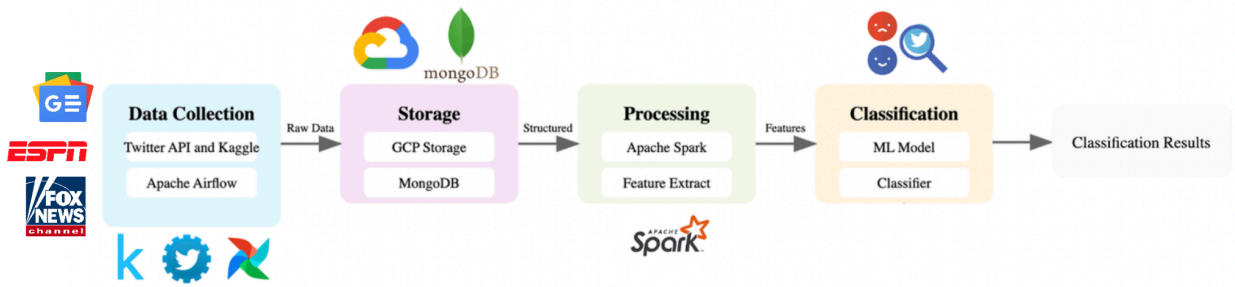
Project Reflection

Story



Reflection

Overall, this was a good practice in implementing software that was new to all of us, and allowed us to understand what a distributed system looks like in application. There are many areas we could improve on in future personal projects, and this project laid a foundation for us to build upon.

- In Task 1, we created a timeline and set goals for the project, making sure each of our members were on the same page before beginning the project. We started with an assortment of datasets and ideas, eventually narrowing our interests to fake vs. real news classification. Many of us were particularly interested in pulling from the Twitter API, but we ran into a problem of the number of pulls needed for a project of this size. Restrictions on data pulls required us to shift to mainly Kaggle and Google News for our data, with a smaller percentage of Twitter than originally anticipated. We established a regular schedule of meetings, every Friday either before or after the seminar, and we stuck to the schedule well.
- We rarely varied from the timeline we created in Task 1, and had a very balanced level of teamwork across members. Many of our group members pursued a AWS or Google Cloud certification in the Fall Module 1 communications class, which helped us in the creation and use of the bucket.
- In Task 2, we performed basic aggregations by news source and by news validity. We then fit a basic Logistic Regression model for Task 3, to accompany our Spark SQL queries that were created by calling data directly from mongodb atlas. Our Kaggle and API data were combined on Mongo and then 100 rows were randomly loaded from the combined_data dataset and used to fit our model. Because it is random, a threshold was set of 50% for accuracy. The function runs for 3 times for better score and breaks. It then loads the test file created from Google News data, randomly picks 50 rows, and predicts whether the news is real or fake and creates a CSV file. This predicted file is loaded back to Mongo.