**Air Traffic scaling, delay- analysis and prediction using PySpark on Hadoop Cluster**

Scaling, Simulation, delay analysis and delay prediction of 2019 US domestic air flights

Mohammed Zoher Guniem Megh Raj Upreti

University of Stavanger, Norway University of Stavanger, Norway

m.guniem@stud.uis.no mr.upreti@stud.uis.no

# ABSTRACT

As more commercial airplanes take off to the sky, data generated from air traffic is rapidly increasing in both size and complexity. According to the International Air Transport Association (IATA), Air travel industry is anticipated to annually grow by 3.5% in the next two decades. This growth raises big questions about how to gather, store, analyze and use the data generated from the those flying cities in the sky.

# KEYWORDS

Hadoop, PySpark, Power BI, Python, Aviation, Scaling, delay, Prediction, Air traffic, Machine learning, Random forest, Logistic Regression, Decision Trees.

# 1 INTRODUCTION

Over the last few decades, air transport is increasing in popularity because of its speed and comfort which eventually increase the traffic in the airspace. With the great increase in air traffic comes a large increase in the demand for airport and airspace capacity. However, airspace and airport capacity cannot keep increasing at a rate necessary to match the rising demand.

During peak hours, the demand for resources in both airport and airspace is at is highest. Some of the most important resources are:

* Good trained human resources.
* Take-off and landing slots.
* Spacious airspace.
* Available Gates at airports.
* Available taxiways and runways.
* Many other factors.

Airspace congestion and flight delays are two of the most important bottlenecks factors that limits the available resources and causes multiple unhealthy side effects on both the operation of air industry and thus the growth of the economy.

It is therefore crucial to have a good trained human resource to manage the airspace and ground operations to avoid leak of capacity in air industry. In this paper we shall investigate the possibilities of using big data technologies to solve such challenges and limitations. Some of these cutting-edge technologies are:

* Apache Hadoop Clustering, which uses the MapReduce programming model and a network distribution to help solving problems related to big data and its demanding computation.
* Apache Spark framework is a cluster-computing framework that offers an interface to program entire clusters with implicit data parallelism and fault-tolerant. In this project we use the powerful python language to work with spark by its popular library PySpark.
* We will also be using some other python libraries and software’s like “basemap”, “matplotlib” and Power BI to further help and assist our research.

# 2 BACKGROUND AND MOTIVATION

# 2.1 Air Traffic Scaling and Simulation

Today’s modern aircrafts are equipped with numerous sensors which measures the performance and different states of each part and system of the aircraft. From the very basic of flying parameters like speed, altitude and location to more detailed data like temperature and pressure from the airplane’s engines and cabin. Data from each airplane is often sent to datacenters on the ground for use in maintenance and troubleshooting. Most airlines also store this same data about each aircraft for later use in analyzation and simulation.

Through the age of aviation, Simulation has had great benefits in training staff to handle the routines, procedures and challenges of the aviation industry. All trainee pilots and air space controllers on multiple levels must spend a big amount of their training in simulation environments. Such simulations enable them to be prepared for daily handling of air traffic that never stops around the clock. Unfortunately, making a realistic simulation model is a big challenge for all software engineers. Simulation software’s can either generate air traffic randomly or by using data from realistic flights. And because the simulation must be as realistic as possible, the first option is not to consider. This makes any simulation software very dependent on data from real life.

The simulation algorithms must have a normal and not complicated flow and low running time, which means that data must be easily scalable with low overhead. In this research, we take advantage of the known capabilities of Hadoop and Spark to design a scaling algorithm that uses data from earlier US domestic flights to provide realistic data for general use in air traffic simulation and analysis.

# 2.2 Delay analysis

Another problem the aviation industry faces is in the delays caused on arrival and departure, these delays cost both the airlines, and passengers millions of dollars every year. A report about the effects of flight delays on economy was given by the United States Joint Economic Committee (JEC) and it has estimated that delayed domestic passenger flights cost the U.S. economy close to $41 billion in 2007 alone. The table in figure 1 below shows the distribution of delay costs between airlines, passenger and other related businesses. Understanding and predicting such delays should help all involved parts to be prepared and thus minimizing the effect of such delays on the economy.

|  |  |  |  |
| --- | --- | --- | --- |
| **Airline Operating Cost** | **Value of Passenger Time** | **Spillover Costs to the Economy** | **Total** |
| $19.1 Billion | $12.0 Billion | $9.6 Billion | $40.7 Billion |

Different analysis software’s can be used to draw a picture on how and what might cause the delays. Since Microsoft launched its first public version of Power BI in 2015, the interests have grown around it because of its user-friendly GUI and flexibility in handling data. Although Power BI has a relatively big collections of different graphs, maps and other visualization tools, loading a big dataset into Power BI will produce a very heavy visualization model which slows the GUI speed and therefore degrade the quality of the user experience.

This is where we can take advantage of the computational power of Hadoop Cluster computation and PySpark methods to easily extract target information about delays from big datasets and use it in fast and flexible power BI models.

# 2.3 Delay Prediction

To take the research in flight delays to the next level, we shall later apply machine learning to predict and expect such delays before they even occur based on what we already know. Hadoop PySpark has some great built-in libraries that helps runs the most popular and efficient machine learning algorithms on our datasets, and based on related work we should then test the accuracy of some popular machine learning algorithms like random forest, decision trees and logistic regression on our dataset.

An accurate machine learning algorithm can be very helpful when working with big datasets and trying to understand the hidden relations between attributes. But as we already know a machine learning algorithm can be highly accurate with one dataset but very much less accurate with another. Therefore, we should run tests where we compare different machine learning algorithms based on earlier related work and calculate their accuracy before moving on and selecting one machine algorithm to work with in the future.

We should also keep in mind that even though a prediction algorithm works with high accuracy today, this does not mean the same algorithm will necessary give high accuracy prediction in the future, this is why it is important to keep a continuous evaluation of the performance of any available machine learning algorithm on our working dataset.

# 2.4 Dataset

# 2.5 Tools & Setup

# 2.6 Related Work

# 3 AIR TRAFFIC SIMULATION

# 3.1 Design

# 3.2 Analysis

# 3.3 Optimization

# 3.4 Implementation

# 4 FLIGHT DELAY ANALYSIS

# 4.1 Design

# 4.2 Analysis

# 4.3 Optimization

# 4.4 Implementation

# 5 FLIGHT DELAY PREDICTION

# 5.1 Design

# 5.2 Analysis

# 5.3 Optimization

# 5.4 Implementation

# 6 EXPERIMENTAL EVALUATION

# 6.1 Air Traffic Simulation

1. **Experimental setup**
2. **Results**

# Delay Analysis

1. **Experimental setup**
2. **Results**

# Delay Prediction

1. **Experimental setup**
2. **Results**

# CONCLUSION

# 8 FURTHER WORK

# REFERENCES