CS5052 Practical 1: Apache Spark

1 Overview

This piece of coursework involves gaining some knowledge and experience by working with Apache Spark in order to analyse a large dataset. **You are expected to have read and understood all the information in this specification at least a week before the deadline. You must contact the lecturers regarding any queries well in advance of the deadline.**

2 Competencies

1. Have a basic understanding of the Python programming language
2. Understand Apache Spark and its components

3 Practical Requirements

You will develop a console-based application in **Python and Apache Spark** in order to analyse a large dataset. The dataset (published by the UK government) describes pupil absence in schools in England from 2006-2018 [1]. The dataset contains over 280,000 records with multiple data recorded, such as the number of schools, number of pupils, the region/local authority, school type, number of authorised absences, etc. The dataset is available in the P1 folder on Studres and also comes with a *data-guidance.txt* file (provided by the UK Government) on understanding the way the data is structured. You should perform an exploration of the dataset beforehand, making sure you fully understand the structure. Also, you will have to think about ways in which you will deal with the various columns, missing/unnecessary data and anything else you may identify in the task. In developing your solution, you may want to begin by processing a specific time period, making sure your solution works correctly, before extending it to work across the full dataset. There are three parts to this task:

Part 1:

To gain a grade of 14.5, you should implement a set of **core features** in order to:

- Read the dataset using Apache Spark.
- Store the dataset using the methods supported by Apache Spark.

- Allow the user to search the dataset by the local authority, showing the number of pupil enrolments in each local authority by time period (year).
  - Given a **list** of local authorities, display in a well-formatted fashion the number of pupil enrolments in each local authority by time period (year).
- Allow the user to search the dataset by school type, showing the total number of pupils who were given authorised absences because of medical appointments or illness in the time period 2017-2018.
- Allow a user to search for all unauthorised absences in a certain year, broken down by either region name or local authority name.
- List the top 3 reasons for authorised absences in each year.

Part 2:

In order to obtain a grade up to 16.5, you must implement the following **three intermediate features**:

- Allow a user to compare two local authorities of their choosing in a given year. Justify how you will compare and present the data.
- Chart/explore the performance of regions in England from 2006-2018. Your charts and subsequent analysis in your report should answer the following questions: Are there any regions that have improved in pupil attendance over the years? Are there any regions that have worsened? Which is the overall best/worst region for pupil attendance?

Part 3:

In order to obtain a grade greater than 16.5, you must implement some more **advanced features**. Part 3 should build on the functionality provided by Part 2 and provide interesting reports and/or insights into the dataset. Interesting can be interpreted any way you see fit but should include some form of advanced analysis (for example, using predictions). Some suggestions are below:

- Visualisation and interaction of the dataset, using external libraries.
- Analyse whether there is a link between school type, pupil absences and the location of the school. For example, is it more likely that schools of type X will have more pupil absences in location Y?
- Provide school recommendations using machine learning for parents looking for a new school for their child in 2023. Based on past data, can you predict which part of the country will provide schools with the best pupil attendance?
- Any other advanced feature of your choosing

4 Deliverables

Submit via MMS a .zip file containing the following by the 24th March 2023:

- The entire source code.
- A README file describing how to run the application.
- Please do not include the dataset with your submission.
- A .pdf report (not more than 3000 words) describing the design, implementation, and any difficulties you encountered. The report must demonstrate any insight obtained by implementing the features in Parts 1-3. **Please include the word count at the end of your report.**
- In particular, it should include:
  - Summary of the core (Part 1), intermediate (Part 2) and advanced (Part 3) features you have implemented. Where appropriate, add some discussion/justification as to why they have been included/your approach for implementation.
  - Include a table which presents an overview of each of the features implemented in Parts 1,2 and 3.
  - Any problems that you encountered and your solutions.
  - Any diagrams/charts you feel are necessary.
  - A reflective summary discussing the lessons learnt and experience that you have gained after finishing this practical.
- A short video clip in a popular format, such as .mp4, demonstrating the execution of your application. This can be a simple 5-minute screen capture with you talking over the video to describe the functionality of your system. Treat it as a video which gives a quick overview of your solution and the functionality it supports.

An important point to note, **only** Apache Spark and Python must be used for this assignment. You **must not use Pandas**. You are however free to use any other external library you want to use to visualise your dataset, build a Python web app, etc.

5 Marking

You will need to not only implement the functionality described but also produce a quality report. Your work will be marked according to the standard mark descriptors in the School. You can find these in the School Student Handbook [3].

6 Lateness

The standard penalty for late submission applies (Scheme B: 1 mark per 8 hour period, or part thereof) [2].

7 Good Academic Practice

The University policy on Good Academic Practice applies [4].

References

[1] HM Government. *Browse our open data, Data Catalogue - Explore education statistics - GOV.UK*. 2022. URL: https://explore-education-statistics.service.gov.uk/data-catalogue.

[2] University of St Andrews. *Assessment — CS Students Handbook*. 2022. URL: https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/assessment.html#lateness-penalties.

[3] University of St Andrews. *Feedback — CS Students Handbook*. 2022. URL: https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/feedback.html#Mark_Descriptors.

[4] University of St Andrews. *Good academic practice — Current Students — University of St Andrews*. 2022. URL: https://www.st-andrews.ac.uk/students/rules/academicpractice/.