STA 9760

Juna Iafelice

Final Project / Spring 2020

1. **Intro Project phase**

YELP REVIEWS

This project consists on a dataset of Yelp’s businesses, reviews, and user data. The datasets will have information about business across 11 metropolitan areas in four countries. The datasets is saved in JSON format. In total there are 8,000,000 user reviews, information on 174,000 business. All the data sets combined exceeds 10GB of data. There are five originals datasets. I have worked only on two of them Business and Review datasets (yelp\_academic\_dataset\_business.json, yelp\_academic\_dataset\_review.json). The main link where you can download the original datasets is on Kaggle site

<https://www.kaggle.com/yelp-dataset/yelp-dataset>

For both datasets (Business and Reviews) I will provide a shortened sample of them attached on the BB. In BB you will find the new optimized python code, the summary document and a file with the most used words in English and an excel file with the ‘Pivot’ Table with the most efficient combinations of performance factors. The performance factors I been working with are Serialization, Compression, Memory management (cache and persist method).

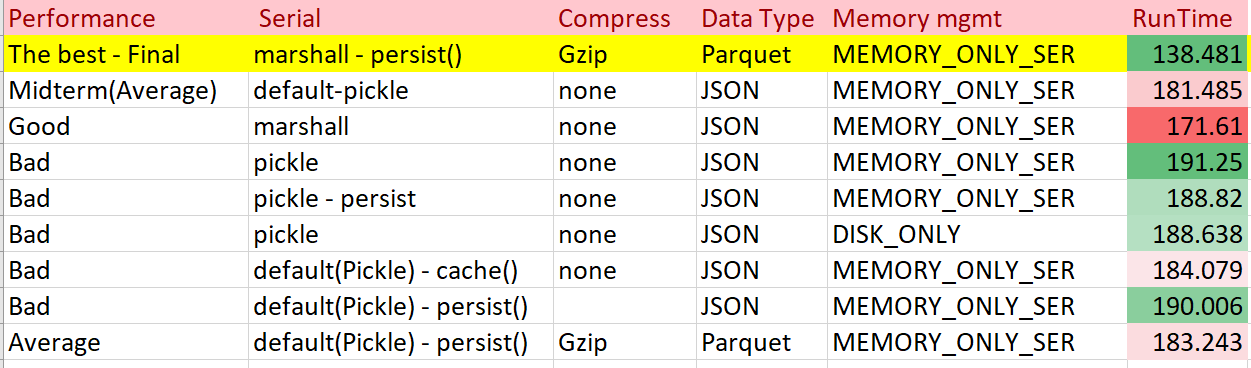
For my final code I converted the JSON data in Parquet data for faster performance. You can find them in the link below. In the link you can find all the project parts as well.

<https://drive.google.com/open?id=1NOdUn6zy8qmqGWS0iNTi-X0QrPVH_Tgq>

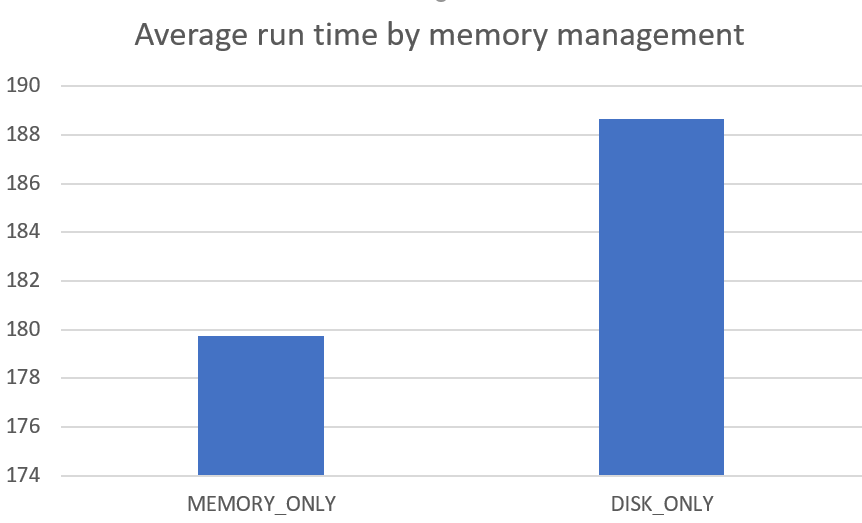
The main question I want to answer in this project is:

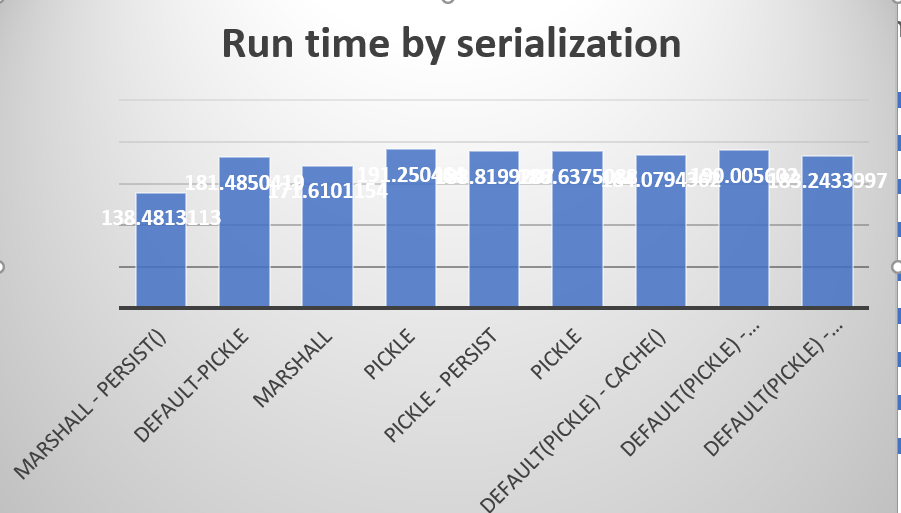
**Question 1: What are the top ten most used words grouped by user star rating reviews?**

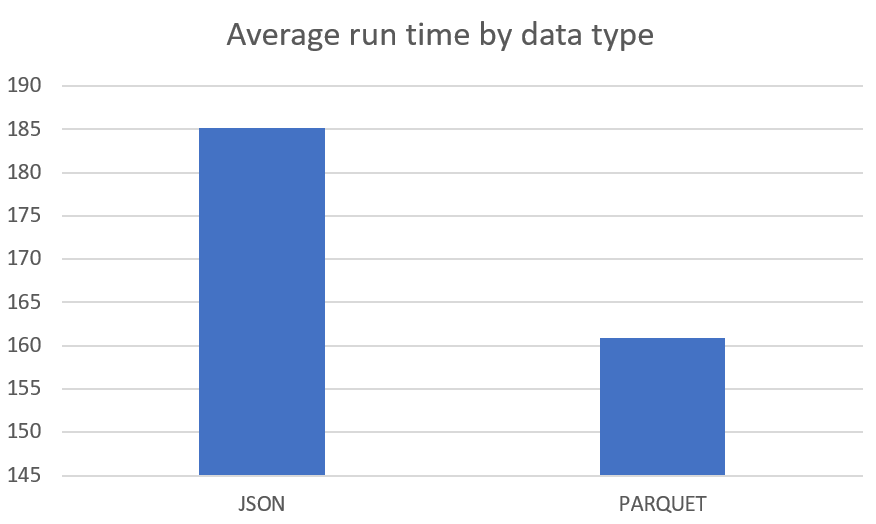
1. **The ‘Pivot’ Table with the most efficient combination of performance factor**



1. **Charts**







1. **Summary**

By using the Parquet compression scheme with in (MEMORY\_ONLY) memory management in combination with Marshal serialization protocol I achieved top performance for my code with a run time of 138.481 sec, a 30% downtime from the midterm run time code. In the context of Spark, I used Marshal Serializer because it is faster but doesn’t support many different data types like pickle. My data type was in JSON but I converted them to Parquet data type because it gave me faster read performance. Also, I compressed them in GZIP format.

Other than this one I tried many Pickle serializations with different optimization technics like cache, persist in MEMORY\_ONLY or persist in DISK\_ONLY. I compressed Jason data types into GZIP, but it didn’t gave my good run time performance. In overall all these other attempts gave me a higher run time performance than my best final optimization as you can see on the ‘Pivot’ table.

You will find attached the best optimizer code with Marshall serialization and the parquet data you will find them on the Google drive provided below.

[**https://drive.google.com/open?id=1NOdUn6zy8qmqGWS0iNTi-X0QrPVH\_Tgq**](https://drive.google.com/open?id=1NOdUn6zy8qmqGWS0iNTi-X0QrPVH_Tgq)