Optimization Report: MongoDB Data Storage and File Generation

Romain LAUP

June 9, 2023

1 Introduction

This report outlines the significant progress made on optimizing our functions for data storage, file generation and user interface. The goal of these improvements was to increase efficiency, and usability.

2 Data Storage and Retrieval

In order to manage the influx of data, functions for storing and retrieving data from our MongoDB database were optimized. This ensures rapid saving of input data and easy retrieval when needed. Additionally, the data processing algorithm is now capable of detecting and categorizing input data into types I, C, or It. When type I data is input, there is also an option to convert it to type J before saving, providing additional flexibility for data management.

3 File Generation

Substantial efforts were put into optimizing the programs that generate Excel and PowerPoint files. Upon uploading the files through the Flask-based user interface, users can now select whether they wish to generate an Excel and/or PowerPoint file. The program autonomously handles the rest, providing an efficient and user-friendly experience.

4 Interface Adaptation

The Flask graphical interface has been adapted to enable seamless interaction with the improved features. This has made it possible for users to effortlessly upload their files and select their preferred output formats. The program's responsiveness and ease of use has seen significant improvement, thanks to these updates.

5 Results and Future Work

The result of these optimizations is a faster processing time, but theses improvements are not sufficient for a big volume of datas. For instance, about 20MB of data takes approximately 1.5 hours to process. Although further optimizations are needed for managing larger volumes of data, the current system processes small volumes of data (less than 1MB) in just a few seconds. Future work will focus on refining these optimizations to further improve the system's performance.