



Search Engine

Stage 2

Data Science and Engineering

Jacob Jażdżyk

Víctor Gil Bernal

Kimberly Casimiro Torres

María Alonso León

Index

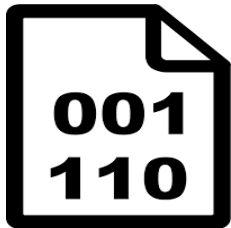
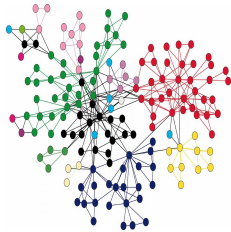
- 1 Introduction
- 2 Objective
- 3 Modules
- 4 Experiments and Tests
- 5 Conclusions
- 6 Future Work

Introduction and Objectives



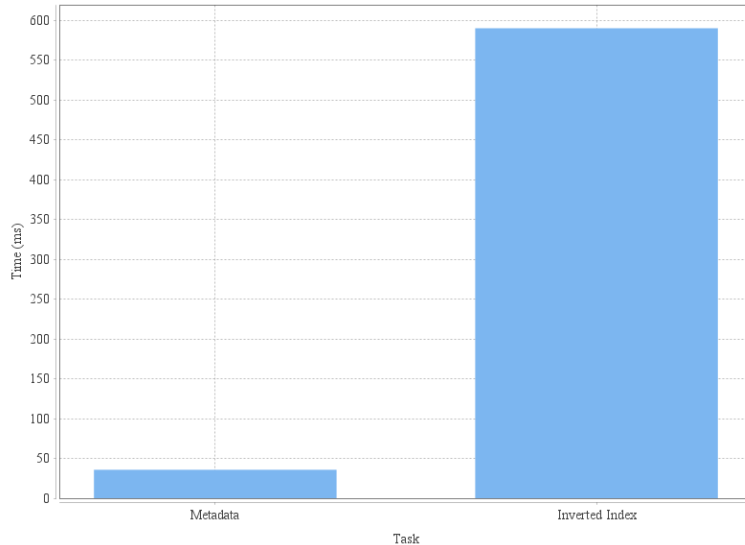
Modules

Indexer

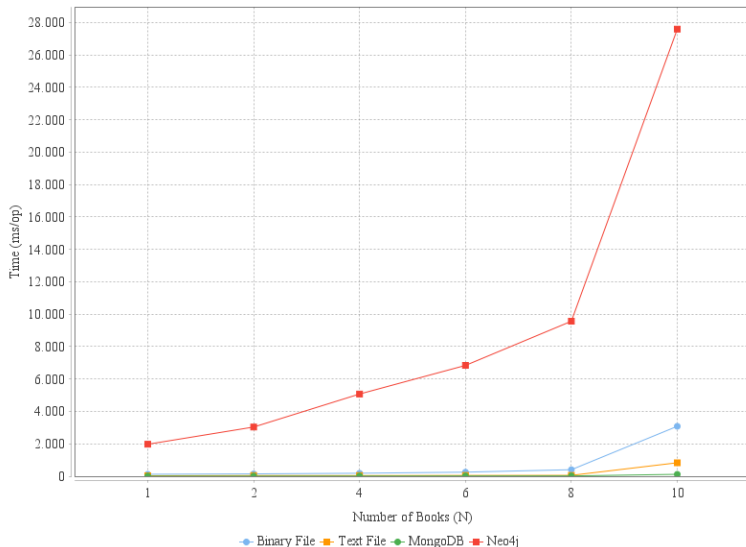


Experiments and Tests

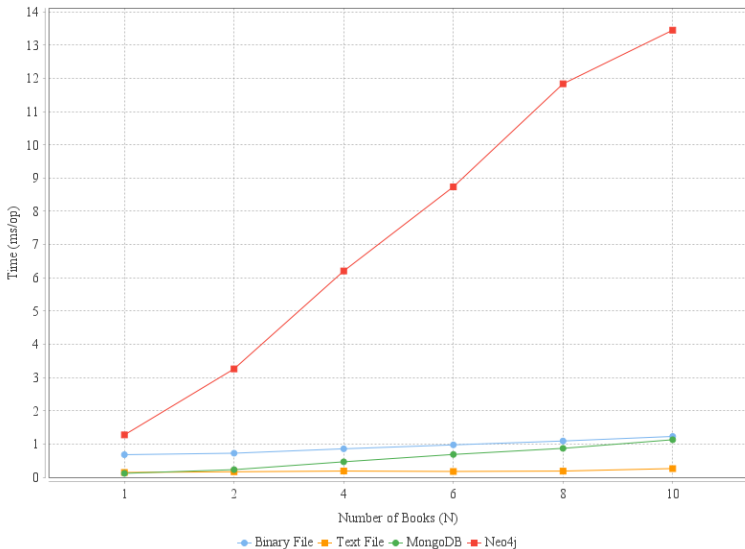
Processing Time Comparison



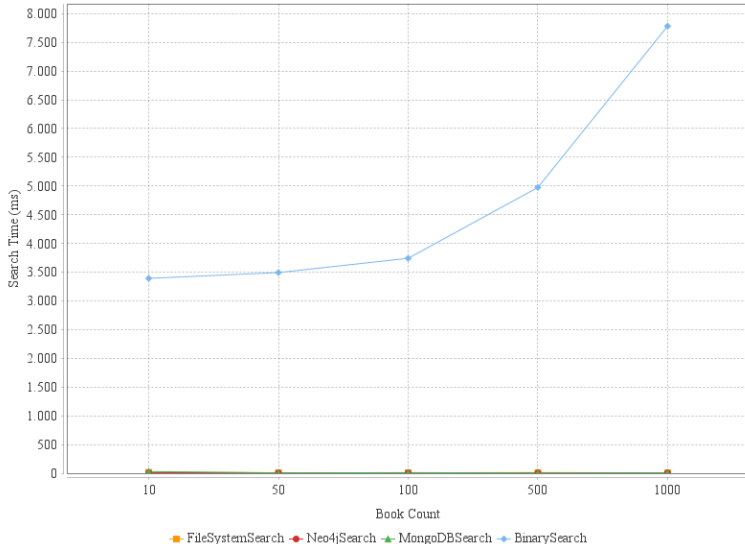
Inverted Index - Processing Time by Storage System



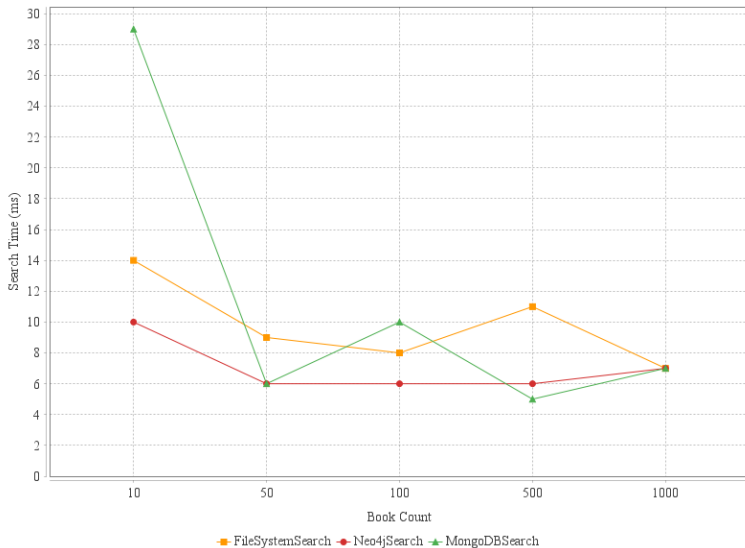
Metadata - Processing Time by Storage System



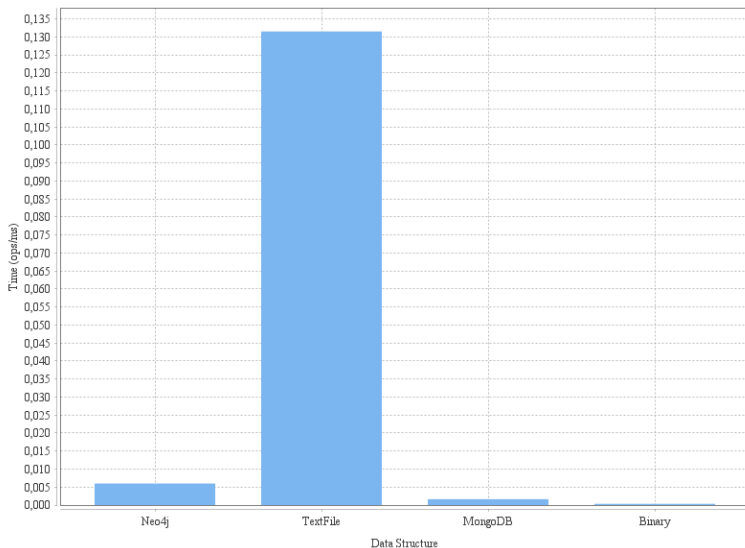
Search Times with Varying Number of Books



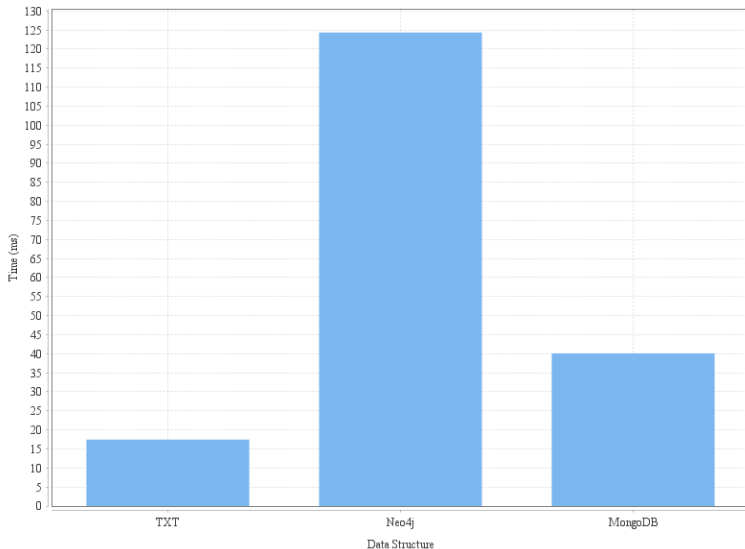
Search Times with Varying Number of Books Without Binary File



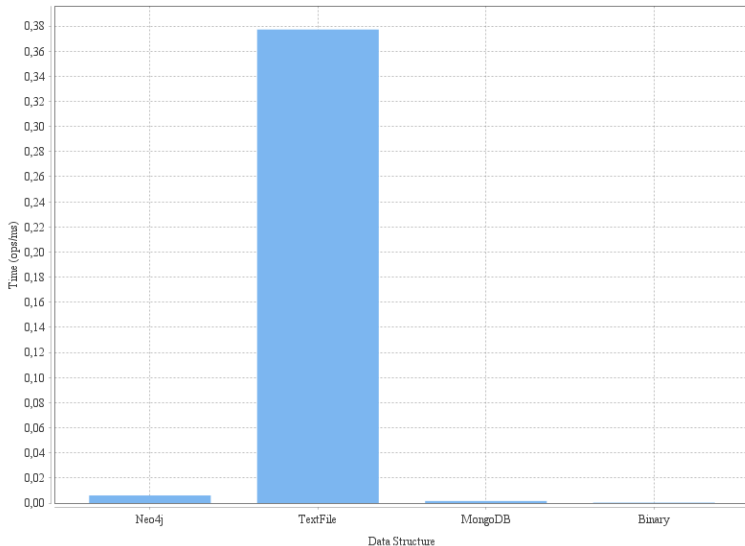
Average Time - Least Frequent Words



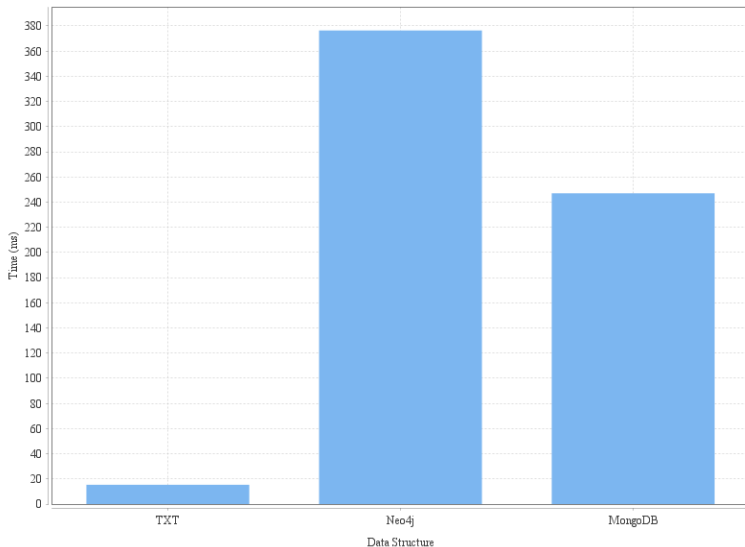
Average Time - Least Frequent Words Without Binary File



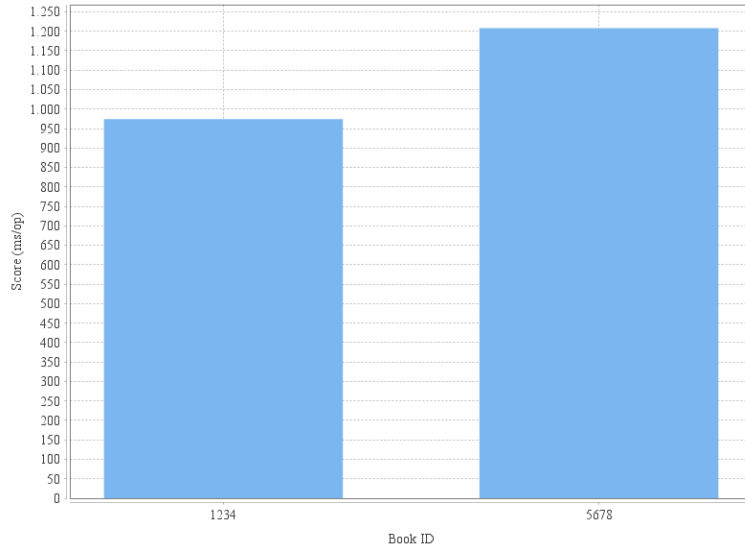
Average Time - Most Frequent Words



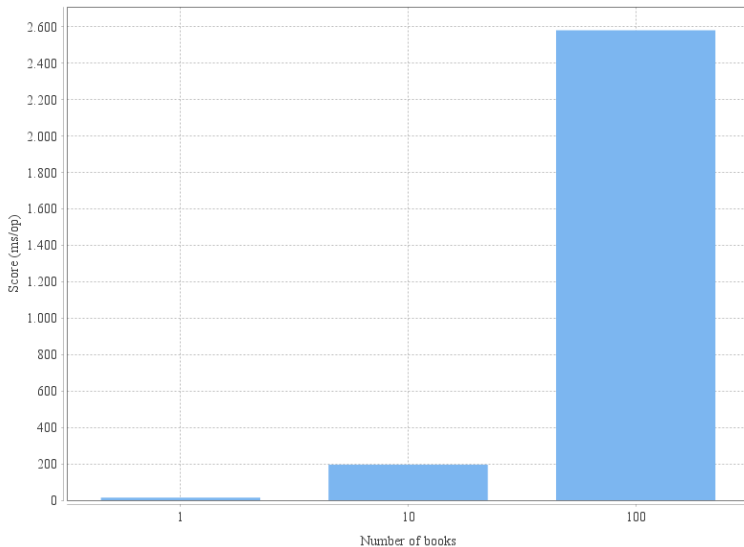
Average Time - Most Frequent Words Without Binary File



Downloading Crawler Performance



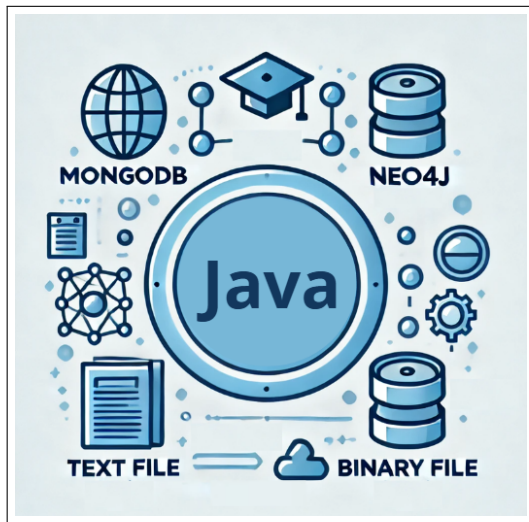
Storage Crawler Performance



Comparison of Python and Java Implementations

Aspect	Python	Java
Language	Easy to learn	Large-scale systems
Design Principles	Dynamic typing, less structured	Strong SOLID adherence
API	FastAPI	Spark
Storage	File System, MongoDB, Neo4j	Binary Files was added
User Interface (UI)	Not implemented	React interface
Performance	Limited scalability	Optimized runtime
Scalability	Challenging	Large data volumes
Flexibility	Rapid changes	Extensible design
Testing and Benchmarking	Pytest	JMH
Deployment	Local environment	Dockerized

Conclusions



Future Work





Thank You