Project Report: Analytics Programming and Data Visualisation

1st Shubham Dalvi
Msc in Data Analytics
National College of Ireland
Dublin, Ireland
x23268051@student.ncirl.ie

2nd Varun
Msc in Data Analytics
National College of Ireland
Dublin, Ireland
x23437391@student.ncirl.ie

3rd George
Msc in Data Analytics
National College of Ireland
Dublin, Ireland
x22248242@student.ncirl.ie

Abstract—**This project focuses on creating a data processing and analysis pipeline using Python and MongoDB. Three datasets—Books, Characters, and Edges—were used, containing both structured and semi-structured data. The data was loaded into MongoDB using Python scripts (*mongo_raw.py*, *mongo_raw_georg.py*, and *mongo_raw_varun.py*). Data cleaning and transformation were carried out with filtering techniques in Python, as implemented in *mongo_retrieval.py*. These steps ensured the data was prepared for analysis and visualization.**

The project analyzed the challenges of handling large datasets, such as cleaning raw data, storing it in a suitable format, and transforming it into structured information. Results from the analysis showed clear relationships within the datasets. Connections between characters and books were visualized to reveal patterns and interactions. These visualizations provided insights into the data and aided in presenting it effectively.

The project demonstrated MongoDB's suitability for semi-structured data and Python's utility in processing and analyzing such datasets. The visualizations offered actionable insights, such as identifying key connections and trends, making the analysis beneficial for both technical and non-technical audiences.

I. Introduction

Understanding relationships within large datasets is critical for effective data-driven decisions. This project explores relationships among books and characters to uncover patterns through structured data processing and visualization. By utilizing Python and MongoDB, the project demonstrates efficient management of semi-structured data, seamless ETL workflows, and impactful visualization techniques.

A. Research Questions

1) How do relationships between characters and books manifest in the dataset?
2) Who are the most influential characters, and what patterns can be inferred from their connections?
3) How can advanced visualization techniques enhance the interpretation of complex networks?

This study aims to extract actionable insights from semi-structured datasets, bridging gaps in narrative analysis using computational tools. By applying systematic methodologies, the project highlights relationships and patterns within literary datasets.

II. Related Work

Existing studies, such as those by Brown et al. (2019), emphasize the value of network visualizations in understanding social structures. MongoDB's effectiveness in managing semi-structured data has been validated in research by Patel et al. (2020). This project integrates Python's ETL capabilities with MongoDB's flexibility, addressing scalability and visualization challenges identified in prior research.

While previous studies focused on structured data or required complex preprocessing for unstructured data, this project leverages modern tools to handle large-scale data and streamline workflows. By incorporating quantitative network analysis, it offers a scalable and reproducible framework for narrative analysis.

III. Data Processing Methodology

A. Datasets

1) **Books Dataset**: Bibliographic records, including identifiers, frequencies, and metadata, formed the foundation for linking characters to narratives and identifying themes.
2) **Characters Dataset**: A catalog of characters with attributes like type (protagonist/antagonist) and role (primary/secondary) enabled the identification of central figures in the network analysis.
3) **Edges Dataset**: Encoded relationships between characters, forming a graph structure. Interaction types (e.g., collaboration/conflict) were used to construct visualizations.

B. Tools and Technologies

- **Programming Language**: Python, chosen for its robust libraries and adaptability in processing large datasets.
- **Database**: MongoDB, ideal for managing semi-structured data with flexible schemas.
- **Libraries**: Pandas, NetworkX, Matplotlib, Plotly, and Seaborn for efficient data processing and diverse visualizations.

C. Workflow

1) **Data Ingestion**: Raw datasets were ingested into MongoDB using Python scripts. This process involved parsing files and ensuring compatibility with MongoDB's structure.
2) **Data Cleaning**: Addressed duplicates, missing data, and inconsistencies using Python scripts. Filtering mechanisms ensured high-quality data for accurate analysis.
3) **Transformation**: Data was normalized into structured formats for easy querying and visualization. This included data type standardization and schema alignment.
4) **Storage and Querying**: Transformed datasets were stored in MongoDB collections and queried for analysis with optimization techniques.

IV. Data Visualization Methodology

A. Visualizations Used

1) **Network Graphs**: Mapped relationships between characters, revealing central figures and clusters. Node sizes were scaled based on degree centrality, emphasizing key influencers.
2) **Bar Charts**: Highlighted the most connected characters and frequent books, enabling comparative analysis.
3) **Heatmaps**: Displayed co-occurrence patterns of characters across books, uncovering significant collaborations and rivalries.
4) **Scatter Plots**: Explored the correlation between character connections and book appearances.
5) **Sankey Diagrams**: Traced character-book relationships to show distribution trends.
6) **Word Clouds**: Highlighted prominent characters based on their frequency in interactions.

B. Design Choices

- **Interactivity**: Tools like Plotly added hover effects and zoom capabilities, enhancing data interpretation.
- **Color and Clarity**: A consistent color scheme was used to distinguish nodes and categories. Warm tones emphasized central nodes, while cooler tones represented peripheral characters.
- **Dashboards**: Combined visualizations into an interactive dashboard, providing a holistic view of the data.

V. Results and Evaluation

A. Key Findings

1) **Central Characters**: Network analysis revealed high-degree nodes representing key influencers pivotal to the narrative.
2) **Cluster Insights**: Community detection algorithms identified tightly knit groups, reflecting subplots or character arcs.
3) **Book Trends**: Recurring books served as hubs for character interactions, underscoring their importance in the dataset.
4) **Character Co-occurrence**: Heatmaps quantified collaborations and rivalries, providing insights into dynamics.
5) **Character Distribution**: Sankey diagrams revealed how characters influenced multiple narratives.
6) **Scatter Plot Insights**: Characters appearing in multiple books often had higher connectivity, showcasing their role as narrative bridges.
7) **Interactive Dashboard Feedback**: Users reported that the dashboard simplified exploration of relationships, making the data accessible to technical and non-technical audiences.

B. Challenges and Solutions

- **Handling Sparsity**: Addressed missing relationships through imputation and filtering, ensuring dataset completeness.
- **Visualization Scalability**: Optimized large networks using community-focused views to maintain clarity.
- **Performance Optimization**: Used query optimization and efficient data structures to reduce computational overhead.

VI. Conclusion

This project demonstrates the effectiveness of Python and MongoDB in managing and analyzing semi-structured datasets. Network analysis and visualization techniques provided actionable insights into character interactions and narrative patterns. The results highlight the importance of structured workflows and modern tools in addressing real-world data challenges.