Advanced Topics in Data Engineering

***Entity Resolution Assignment***

***Supervisor: Mr. Giorgos Alexiou***

A picture containing dark, high

Description automatically generated

**Thursday August 31st, 2023**

**Dimitris Matsanganis, f2822212**

# Abstract

This assignment focuses on the intricate domain of entity resolution and duplicate detection, utilizing advanced methodologies to augment data precision and integrity. Through a structured progression, each task contributes to a comprehensive comprehension of the methodologies' nuances.

Task A: Token Blocking for Block Creation

Task A employs the Token Blocking approach, schema-agnostic in nature, to construct blocks represented as Key-Value (K-V) pairs. All attributes except the identifier (id) contribute to generating distinct Blocking Keys (BKs) from entity attribute values. The id column remains non-participatory in the block index creation process. Lowercasing string attributes during tokenization ensures precise alignment. A function is employed to visualize the BKs in a reader-friendly manner.

Task B: Calculating Possible Comparisons

In Task B, we compute the full spectrum of necessary comparisons to rectify duplicates within the established blocks from Task A. The quantified number of comparisons offers insights into the computational intricacies of the entity resolution process.

Task C: Meta-Blocking Graph with CBS Weighting Scheme

Task C introduces the Meta-Blocking graph, implementing the CBS (Common Block Scheme) Weighting Scheme. Edges with a weight below 2 are pruned, refining the block collection and eliminating unnecessary comparisons. The pruned collection serves as the foundation for recalculating the ultimate number of comparisons, paralleling Task B's methodology.

Task D: Jaccard Similarity Function

Task D encompasses a custom function for computing Jaccard similarity, focusing on the "title" attribute. While not involving practical comparisons, this function serves as a gauge for attribute similarity.

This analysis is executed in Python, specifically version 3.10, using Jupyter Notebook. The assignment amalgamates theoretical understanding with hands-on application, fostering mastery of advanced entity resolution techniques and bolstering data analysis proficiencies.

# Contents

[Abstract 2](#_Toc142433034)

[Contents 3](#_Toc142433035)

[Table of Figures 3](#_Toc142433036)

[Assignment Description 4](#_Toc142433037)

[Case Study Outline 5](#_Toc142433038)

[I. Introduction 7](#_Toc142433039)

[II. Task A: Token Blocking for Block Creation 8](#_Toc142433040)

[III. Task B: Calculating Possible Comparisons 8](#_Toc142433041)

[IV. Task C: Meta-Blocking Graph with CBS Weighting Scheme 8](#_Toc142433042)

[V. Task D: Jaccard Similarity Function 8](#_Toc142433043)

[VI. Conclusion 8](#_Toc142433044)

[VII. Deliverables 8](#_Toc142433045)

[VIII. Appendix 8](#_Toc142433046)

# Table of Figures

[Figure 1: Installation and Setup of Impala via Cloudera QuickStart VM. 5](#_Toc142416818)

[Figure 2: Create the Impala database & the required tables. 9](#_Toc142416819)

[Figure 3: Provided schema. 10](#_Toc142416820)

[Figure 4: Executing the Task B's command variable type error. 10](#_Toc142416821)

[Figure 5: Create the Impala database & the required tables (revisited version). 11](#_Toc142416822)

[Figure 6: Validation procedure of task 3a. 12](#_Toc142416823)

[Figure 7: Executed command successfully adds student record in Impala. 13](#_Toc142416824)

[Figure 8: Execution of Task 3c's query in Impala with Results. 15](#_Toc142416825)

[Figure 9: Execution of Task 3d's query in Impala with Results. 17](#_Toc142416826)

[Figure 10: Results of explain command for demonstrative purposes. 18](#_Toc142416827)

[Figure 11: A sample picture demonstrating the procedure followed for the Task 4. 20](#_Toc142416828)

# Assignment Description

**Task A [30 points]**

Use the Token Blocking (not to be confused with Standard Blocking) method as a schema-agnostic approach to create blocks in the form of K-V (Key-value) pairs. The key for every entry will be each distinct Blocking Key (BK) derived from the entities' attribute values, except for the id column which should ONLY be used as a reference for the blocking index and should NOT be included in the blocking process (block index creation). Please note that Token Blocking, being schema-agnostic, allows the utilization of all attributes from every entity (except id) for creating the blocks. To ensure accurate matching, it is advised to transform every string to lowercase during the tokens' creation before inserting them into the index. At the end of the creation, use a function to pretty-print the index, displaying the Key-Value pairs in a clear and readable format.

By employing the Token Blocking method in a schema-agnostic manner, you will create blocks that capture the essence of the entities' attributes. Each block's key will represent a distinct Blocking Key (BK) derived from these attribute values, excluding the id column. This approach allows for comprehensive analysis by utilizing all available attributes from each entity. Lowercasing the attribute strings during token creation helps avoid mismatches. Finally, ensure the generated index is presented neatly using a function to pretty-print the Key-Value pairs.

**Task B [25 points]**

Compute all the possible comparisons that shall be made to resolve the duplicates within the blocks that were created in Task A. After the computation, please print the final calculated number of comparisons.

**Task C [30 points]**

Create a Meta-Blocking graph of the block collection (created in Task A) and using the CBS Weighting Scheme (i.e., Number of common blocks that entities in a specific comparison have in common) i) prune (delete) the edges that have weight < 2 ii) re-calculate the final number of comparisons (like in step B) of the new block collection that will be created after the edge pruning.

**Task D [15 points]**

Create a function that takes as input two entities and computes their Jaccard similarity based on the attribute title. You are not requested to perform any actual comparisons using this function.

# Case Study Outline

In this case study, we embark on a comprehensive exploration of advanced techniques for entity resolution and duplicate detection. In today's data-driven world, the importance of ensuring data accuracy and quality cannot be overstated. Duplicates lurking within datasets can taint analytical outcomes and introduce erroneous insights. The advanced methodologies uncovered in this case study furnish a robust framework to surmount these challenges and elevate data precision to a new level.

The systematic journey unfolds across four distinct tasks, each contributing to the development of a holistic understanding of entity resolution methodologies. These tasks guide us through the intricate landscape of data cleansing and entity consolidation, armed with techniques that stand at the forefront of data refinement. From the art of creating data blocks through Token Blocking, and the quantification of comparisons to efficiently resolve duplicates, to crafting Meta-Blocking graphs and gauging attribute similarity, this case study equips us with the precise tools essential for navigating complex data scenarios.

At the heart of this case study lies the integration of theoretical knowledge with practical implementation. Python 3.10, a powerful and versatile programming language, collaborates seamlessly with the dynamic environment of Jupyter Notebook. Together, they form the vessel for executing and demonstrating the advanced techniques outlined in this study. The ensuing sections of this case study unravel the intricate threads woven into the fabric of entity resolution and duplicate detection, inviting readers to immerse themselves in the richness of these methodologies.

**Assignment Structure:**

**I. Introduction**

The foundation is laid with an exploration of data quality's paramount importance. Within the expansive realm of data management and analysis, the specter of duplicates threatens accuracy. This assignment's core objective revolves around the investigation of advanced methodologies that elevate data integrity through precise identification and resolution of duplicates.

**II. Task A: Token Blocking for Block Creation**

The initial step introduces us to the Token Blocking method—an avant-garde schema-agnostic technique. By leveraging all attributes except the identifier (id), we construct Key-Value (K-V) pairs that encapsulate distinctive Blocking Keys (BKs). These BKs, derived from entity attribute values, become the building blocks for data consolidation. The id column's passive role in block index creation is emphasized, alongside the importance of lowercasing string attributes for precise tokenization. The result is a structured index, elegantly visualized using a dedicated function.

**III. Task B: Calculating Possible Comparisons**

In Task B, we tackle the computational intricacies entailed in resolving duplicates. The task entails a comprehensive calculation of all requisite comparisons within the established blocks from Task A. By quantifying the magnitude of computations required, we gain insight into the computational complexity inherent in precise entity resolution.

**IV. Task C: Constructing a Meta-Blocking Graph**

Task C introduces us to the concept of Meta-Blocking, a powerful strategy that leverages the CBS (Common Block Scheme) Weighting Scheme. Assembling a graph based on the block collection from Task A, we prune edges with weights below 2—a strategic move to streamline the block collection and curtail superfluous comparisons. The pruned collection forms the bedrock for recalculating the definitive number of comparisons.

**V. Task D: Jaccard Similarity Function**

Task D unveils a custom-crafted Jaccard similarity function, tailor-made for attribute comparison based on the "title" attribute. Though devoid of practical comparisons, the function serves as an invaluable tool for quantifying attribute similarity—a linchpin in entity resolution.

**VI. Conclusion**

As the case study culminates, the accomplishments of each task and the methodologies' intricacies converge. The assignment concludes with a reflective summary, shedding light on the multifaceted contributions of executed techniques to bolstered data quality, encapsulating the essence of accurate entity resolution and duplicate detection.

**VII. Deliverables**

The assignments and analysis deliverables will be mentioned in this section.

**VIII. Appendix**

Delve further into the world of advanced techniques with these supplementary resources. Explore Token Blocking, Meta-Blocking, Jaccard similarity, Python, Jupyter Notebook, and recommended reading for a well-rounded understanding of entity resolution.

# I. Introduction

In the intricate realm of **data management and analysis**, the **cornerstone of success** lies in unwavering **data quality** and unwavering **accuracy**. Amidst this pursuit, an **omnipresent challenge** emerges in the form of **duplicates** residing within datasets. These duplicates possess the **uncanny ability** to distort insights and introduce misinterpretations, demanding the application of nothing short of **robust and advanced techniques** for effective **entity resolution** and meticulous **duplicate detection**.

Within the **purview** of this assignment, the **profound importance** of **data quality** takes center stage. Duplicates, when left unchecked, can inflict **havoc** upon analytical endeavors, casting a **shadow of doubt** upon the **credibility** of outcomes. To address this **critical concern**, this assignment embarks on a journey to unravel a suite of **advanced methodologies** meticulously crafted to not only detect but to **precisely identify** and rectify duplicates.

Our **voyage of exploration** commences with **Task A**, where the innovative **Token Blocking method** is introduced. This **schema-agnostic approach** ingeniously creates **blocks** through the harmonious pairing of **Key-Value (K-V) pairs**. These pairs, stemming from distinctive **Blocking Keys (BKs)**, encapsulate the very **essence** of entity attributes. However, the **id column** stands distinct, reserved solely for referencing within the **blocking index**, thus serving as a marker but not actively participating in the block creation process. An important note to consider is the transformation of **string attributes** to **lowercase** during token creation, ensuring meticulous accuracy and circumventing **mismatches**. The culmination of this process is a **visually appealing** and **intelligible Key-Value pair representation**, presented through a dedicated **function**.

Advancing forward, **Task B** ventures into the domain of **computational complexity**. Here, we undertake the meticulous **quantification** of all **conceivable comparisons**. This calculation shines a **spotlight** on the intricate web of comparisons required for the **precise resolution of duplicates** within the blocks that were methodically constructed in **Task A**.

As our exploration unfolds, **Task C** unveils the realm of **Meta-Blocking**, introducing a thoughtfully constructed **graph**. This graph is birthed from the **block collection** forged in **Task A**, and it harnesses the potency of the **CBS (Common Block Scheme) Weighting Scheme**. Pruning edges with weights below the threshold of **2** stands as a **strategic decision**, streamlining the block collection and minimizing the **quantum of superfluous comparisons**. This conscientious curation forms the **bedrock** for recalculating the ultimate **number of comparisons**.

**Task D**, the final stride, delves into the creation of a function that mirrors the **Jaccard similarity concept**. This custom function, designed to compute similarity based on the attribute “**title**,” emerges as an **indispensable tool** for gauging the **likeness of attributes**. Although this function does not perform **actual comparisons** within this assignment, it is poised to play a **pivotal role** in evaluating similarity across the **expanse of attributes**.

This assignment unfurls within the canvas of **Python 3.10** and the dynamic tapestry of **Jupyter Notebook**. United in purpose, these tools merge **theoretical understanding** with **hands-on application**. As we journey through these tasks, let’s embrace the **opportunity** to **hone our skills**, cultivate a **profound understanding** of **advanced techniques** for entity resolution, and contribute to the realm of **data integrity** and **accuracy**. Through this **comprehensive voyage**, we shall refine our **expertise**, bridging the gap between **theory and practice**, and ultimately elevate our capability to wield **advanced methodologies** for meticulous **entity resolution** and **duplicate detection**.

# II. Task A: Token Blocking for Block Creation

# III. Task B: Calculating Possible Comparisons

# IV. Task C: Meta-Blocking Graph with CBS Weighting Scheme

# V. Task D: Jaccard Similarity Function

# VI. Conclusion

# VII. Deliverables

# VIII. Appendix