Analysis of Similarity Join Algorithms

Keywords: similarity join, string comparison, parallel computation

Problem:
String similarity joins have been widely used to detect duplicates and data inconsistencies. Thus, literature involves multiple different algorithms and string similarity metrics. However, depending on the workload, some similarity join algorithms might behave better in terms of efficiency and accuracy, e.g., splitting large strings into tokens might hurt performance. In order to facilitate the similarity join process, there is need for an analysis that determines how different algorithms behave depending on the given workload.

Project: Perform an analysis of scale-out string similarity join algorithms given different synthetic and real-world workloads. Specifically, implement and evaluate unsupervised and supervised algorithms involving string tokenization, string clustering, word embeddings using efficiency and accuracy as metrics. The workload will involve datasets with different characteristics, such as different string complexity and different levels of similarity among the input strings.

Plan:

1. Get acquainted with the literature
2. Implementation of unsupervised and supervised string similarity join algorithms in Spark
3. Construction of different data workloads
4. Evaluation of each algorithm or combination of algorithms
5. Analysis of the results for each workload

Supervisor: Prof. Anastasia Ailamaki, anastasia.ailamaki@epfl.ch
Responsible collaborator(s): Stella Giannakopoulou, stella.giannakopoulou@epfl.ch
Viktor Sanca, viktor.sanca@epfl.ch

Duration: 3 months