Work Progress

kNN Search with Parallel Incremental Query Answering

Jaouhara Chanchaf

Tuesday Dec. 06th, 2022

1. Summary

Done:

AI 1	Find a spell checker for PDF documents.
AI 1	Kashif PQA: Fix bug in increment barrier.
AI 4	PEXEO: run experiments using the same settings in the paper.
AI 5	Kashif: Stop when NN distance changes. Measure recall based of results from ~~LSH ensemble~~ and PEXESO.

In progress: Improve PEXESO performance.

Not started:

AI 6	Pick a query vector and manually label accurate NN. Measure recall and visualize the correlation between the NN accuracy and distance to the query vector.
AI 2	LSH Ensemble: get familiar with the code and run it on WDC.

2. Performance

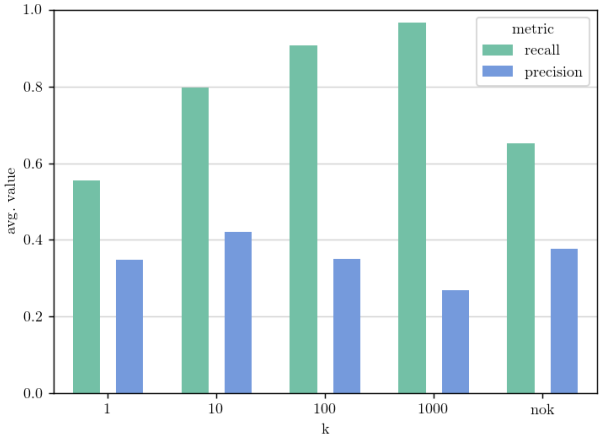

Figure: Kashif average recall and average precision

- **Dataset:** 1000 tables - 6,337 columns - 735,011 vectors.
- **Query:** 10 Queries of size 100.
- **Ground truth:** PEXESO, default settings: $\tau = 6\%$, $T = 60\%$

2. Performance

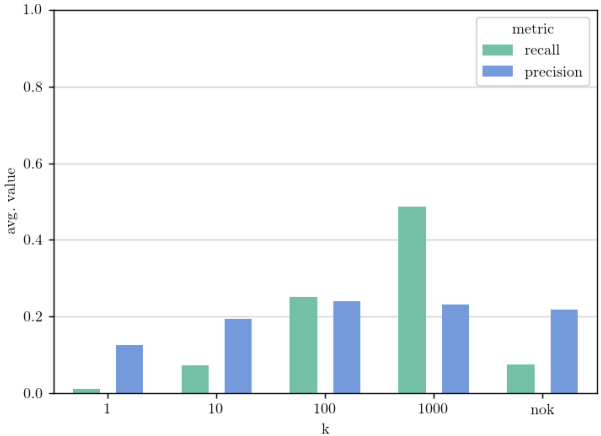

Figure: Kashif average recall and average precision

- **Dataset:** 1000 tables - 6,337 columns - 735,011 vectors.
- **Query:** 10 Queries of size 100.
- **Ground truth:** PEXESO, settings: $\tau = 6\%$, $T = 1\%$

2. Performance

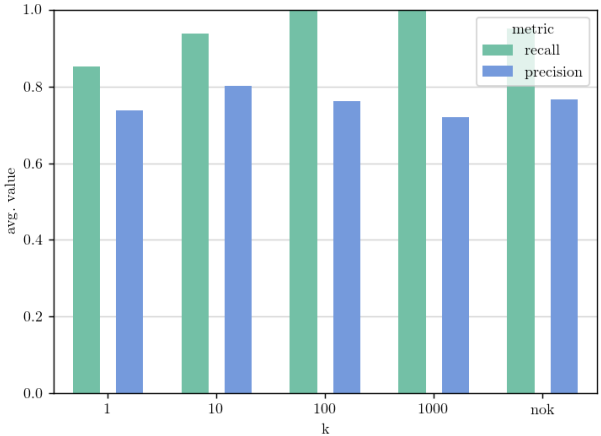

Figure: Kashif average recall and average precision

- **Dataset:** 1000 tables - 6,337 columns - 735,011 vectors.
- **Query:** 10 Queries of size 100.
- **Ground truth:** PEXESO, settings: $\tau = 2\%$, $T = 20\%$

3. Discussion

- Cannot run PEXESO on larger datasets, must optimize implementation.
- Cannot compare Kashif results with LSH ensemble results. LSH ensemble performs table union and not table join.