Universität Bamberg

Examining "Coresets over Multiple Tables for Feature-rich and Data-efficient Machine Learning“ from Wang et al. [1]

Julius Stutz

Outline

Problem – Feature-rich ML

- Problem:
  Feature-rich Machine Learning takes a long time

- Solution:
  Use coreset of original data to reduce time needed

Problem – Coreset selection

- Problem:
  Current coreset selection algorithms work with one table, what if we have many different tables?

- Solution:
  Joins tables together

Problem – Join materializations

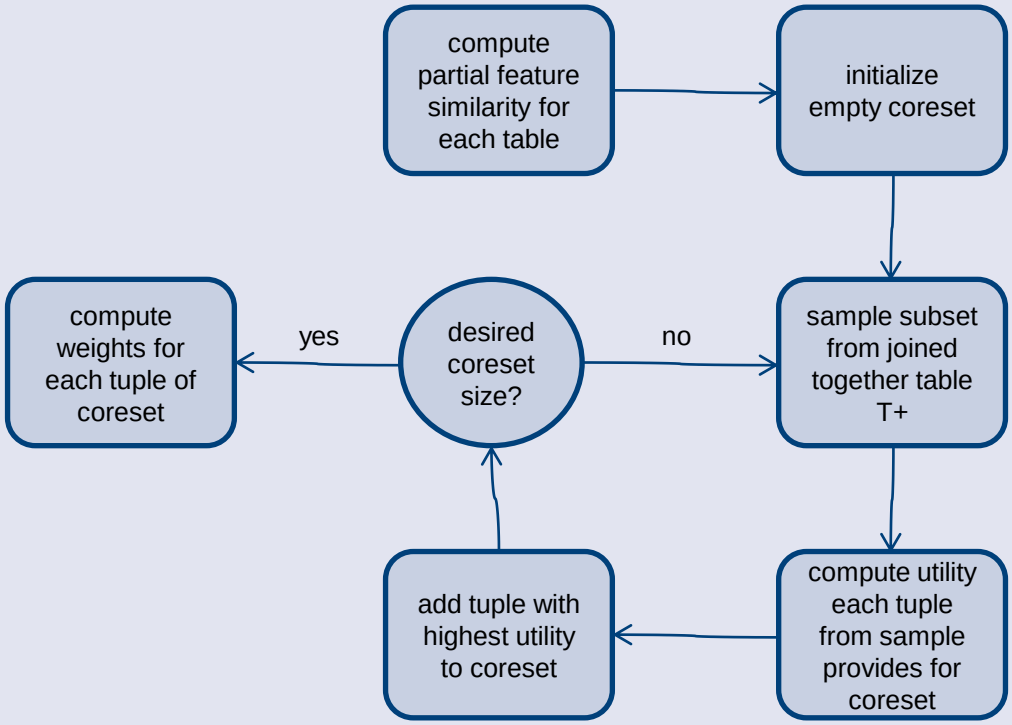
- Problem:
  Joining multiple tables with lots of data each together is computationally expensive

Solution – RECON algorithm

- RECON = feature-**R**ich and data-**E**fficient machine learning with **C**oreset selection with**O**ut join materializatio**N**

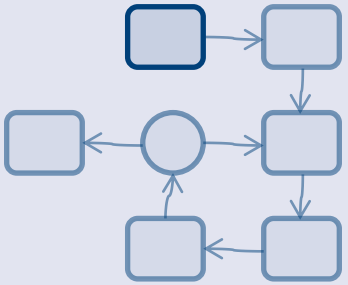
- Concept:
  Select coreset without joining all tables together by calculating how representative a tuple is for the respective table

Solution – RECON algorithm



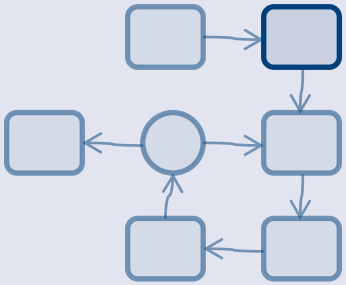
Solution – compute partial feature similarity

- partial feature similarity = degree of similarity of two tuples in a table
- partial feature similarity computation done for each table (as a pre-computation step)



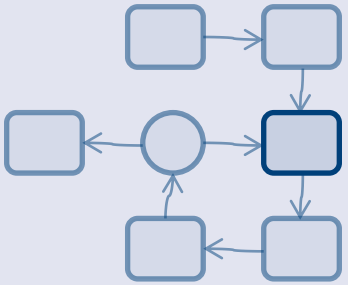
Solution – initialize empty coreset

- create coreset as the empty set
- following loop will continue till desired coreset size is reached



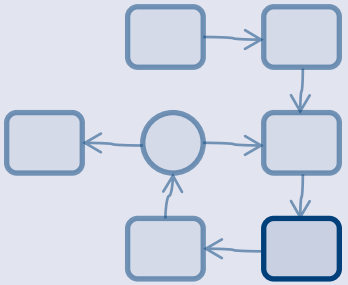
Solution – sample subset

- join materializations should not happen
- thus a small (uniform [2]) sample is needed to obtain potential values to add to the coreset
- sample is subset of tuples of joined together table T+



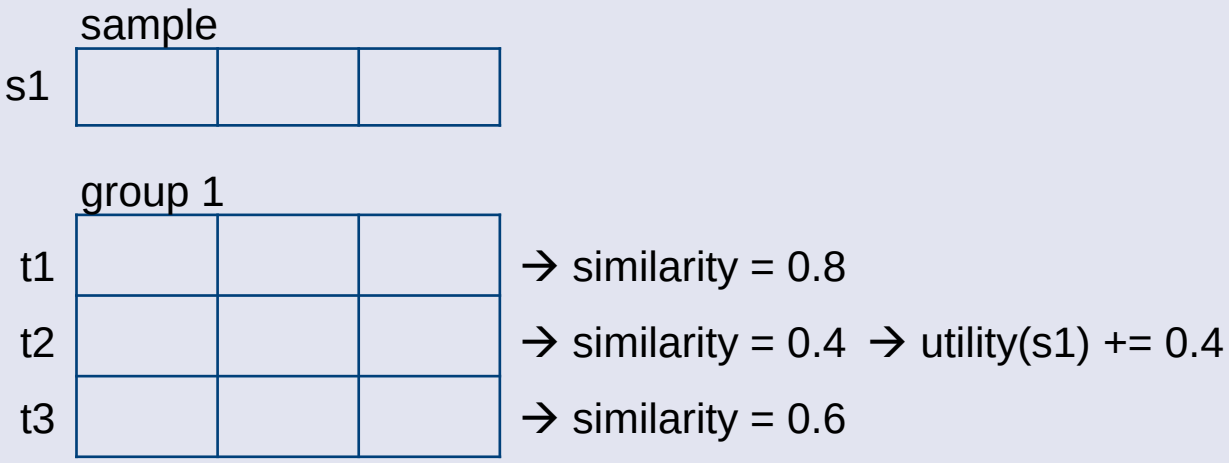
Solution – compute utilities for coreset

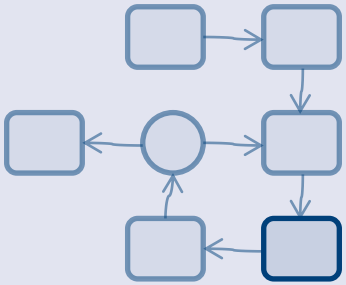
- goal: compute utility each tuple in sample provides for coreset
- utility for each tuple initially set to zero
- then all tables are divided into disjunct groups according to a group key (tuples with the same values for the group key are in the same group)



Solution – compute utilities for coreset

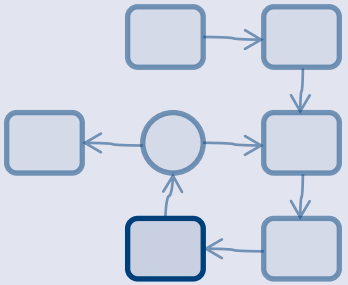
- utility of each tuple in sample computed by comparing it to each group, taking the minimal degree of similarity found in the group (worst case), and adding it to current utility





Solution – add tuple to coreset

- when utility for each tuple in sample is computed: add tuple with highest utility to coreset

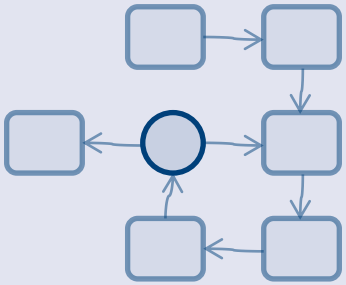


Solution – check coreset size

- desired coreset size reached?

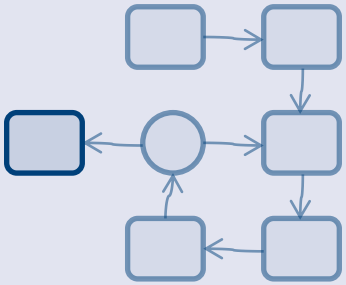
→ no:
return back to loop by sampling new tuples for T+ (right arrow)

→ yes:
continue with weight computation (left arrow)



Solution – compute weights

- each group is uniquely mapped to a tuple in the coreset
- weight(cx) = ∑ (size of group mapped to cx)
- shows how representative tuple is for whole dataset



Solution – time complexity

- normal coreset selection:
  O ($N^2D + NKS$) N = size of joined table T+

- RECON:
  O ($n^2d + nKS$) n = average size of all single tables

- $N >> n$
  (especially with Feature-rich ML)

Results – compared methods

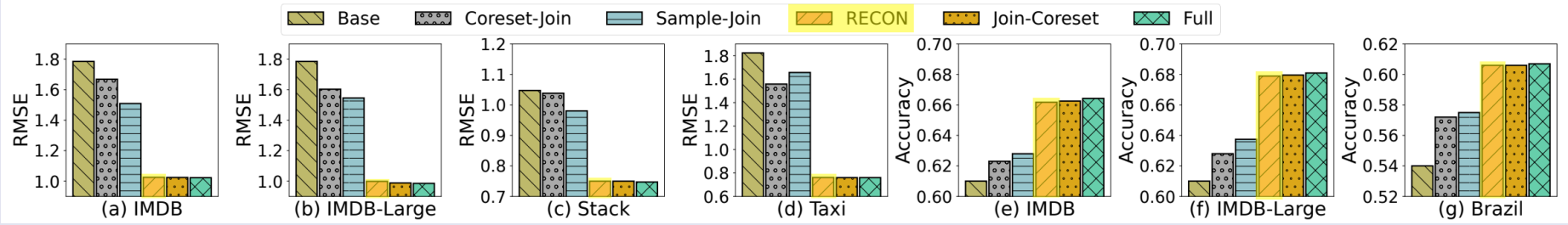
- `Base` (train model only with base table T)
- `Full` (train model with join together table T+)
- `Sample-Join` (train model with sampled T+ data (no joins))
- `Join-Coreset` (train model with coreset of T+)
- `Coreset-Join` (train model by first creating coreset of T and then joining it with other tables)

Results – measurements

- effectiveness:
  - accuracy (for classification tasks)
  - root mean squared error (RMSE) (for regression tasks)

- efficiency:
  - time spent to create coreset (if necessary) and train model

Results – effectiveness



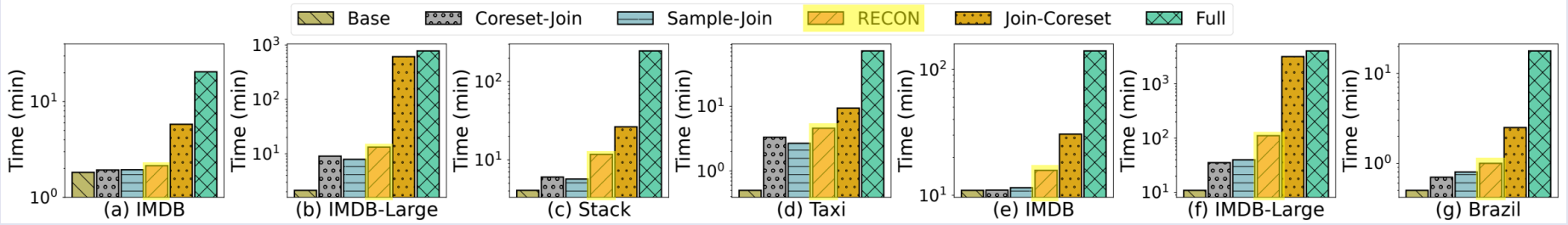
regression tasks

classification tasks

RMSE
(lower is better)

accuracy
(higher is better)

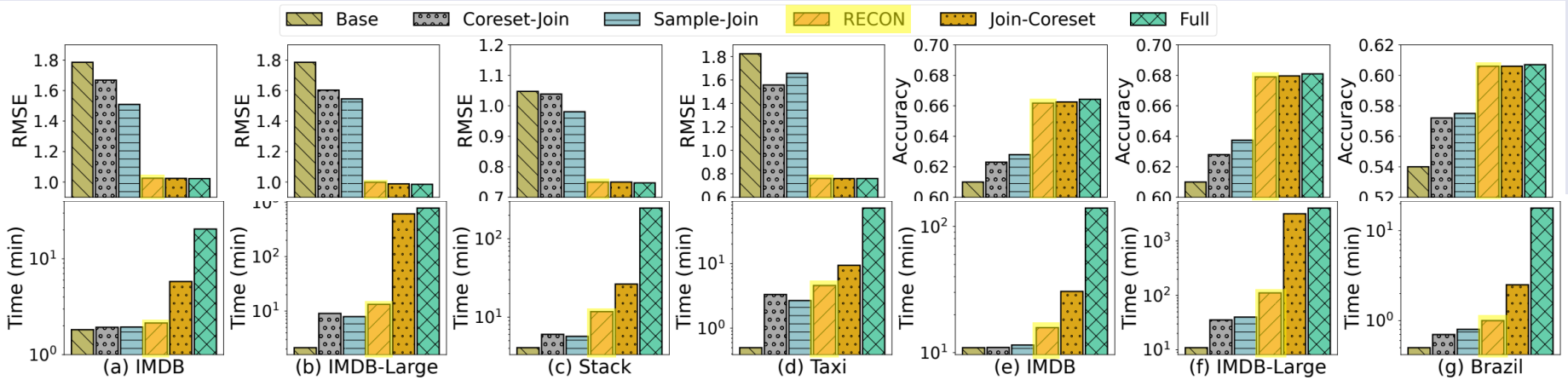
Results – efficiency



regression and classification tasks

time spent to create coreset (if necessary) and train model (lower is better)

Results – combined



`Base`, `Coreset-Join`, `Sample-Join`: good efficiency, bad effectiveness

`Join-Coreset`, `Full`: good effectiveness, bad efficiency

`RECON`: good effectiveness, good efficiency

Bibliography

- [1]: Jiayi Wang, Chengliang Chai, Nan Tang, Jiabin Liu, and Guoliang Li. Coresets over Multiple Tables for Feature-rich and Data-efficient Machine Learning . PVLDB, 16(1): 64 - 76, 2022.

- [2]: Zhuoyue Zhao, Robert Christensen, Feifei Li, Xiao Hu, and Ke Yi. 2018. Random Sampling over Joins Revisited. In SIGMOD 2018. ACM, 1525–1539.