Pythia

A Neural Model for Data Prefetching

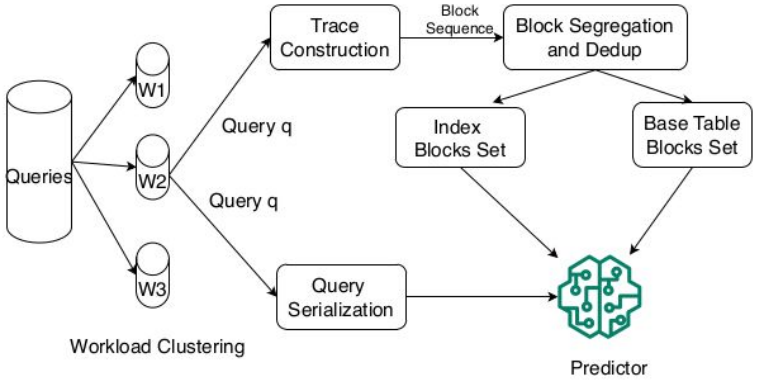
Akshay Arun Bapat, Saravanan Thirumuruganathan, Nick Koudas

Motivation

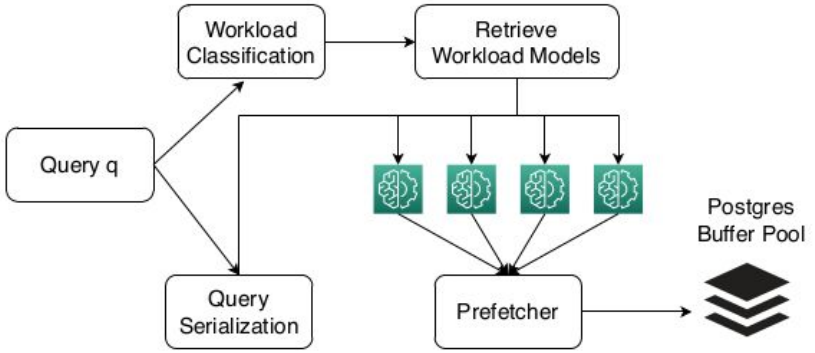
- ML in Databases
  - Cardinality estimation
  - Query optimization
  - Indexing
  - Buffer Manager?

- Instance Optimized
  - Offers correlation (predictability) with specific
    - Workload
    - Data distribution
  - Performance boost

Problem / Goal



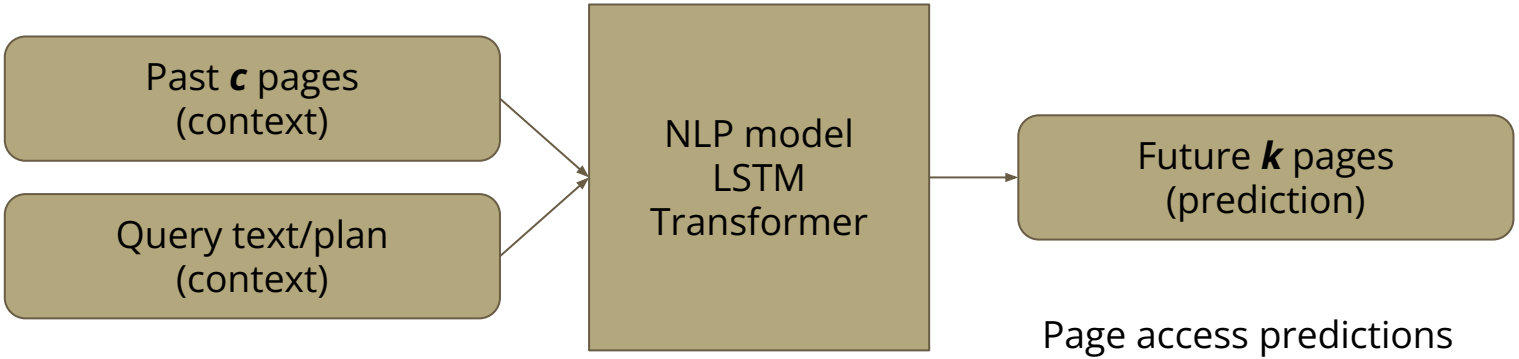
Preprocess



Use

Page access prediction

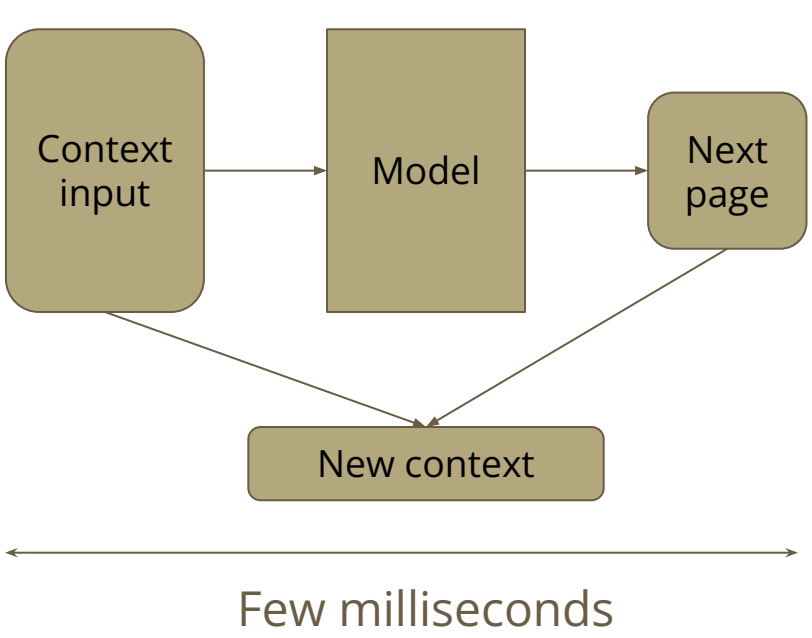
- Predicting access patterns
  - Memory access
  - File access
- Access patterns are a sequence
  - Natural sequence prediction problem
  - NLP models like LSTM and transformers



Transformers do NOT work!!

Transformers do not work!!

- Sequential nature of prediction

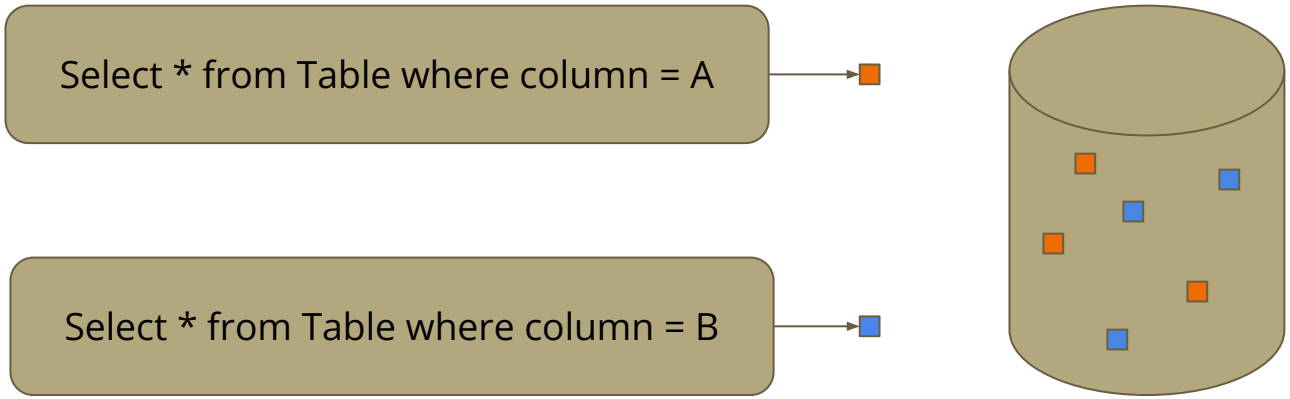


Million times

Impractical for prefetch

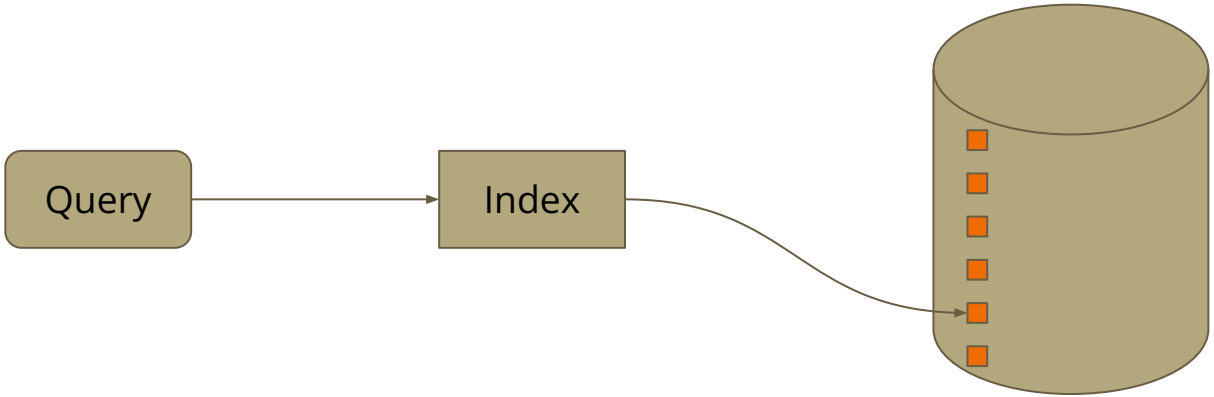
Transformers do not work!!

- Sensitivity to query predicates



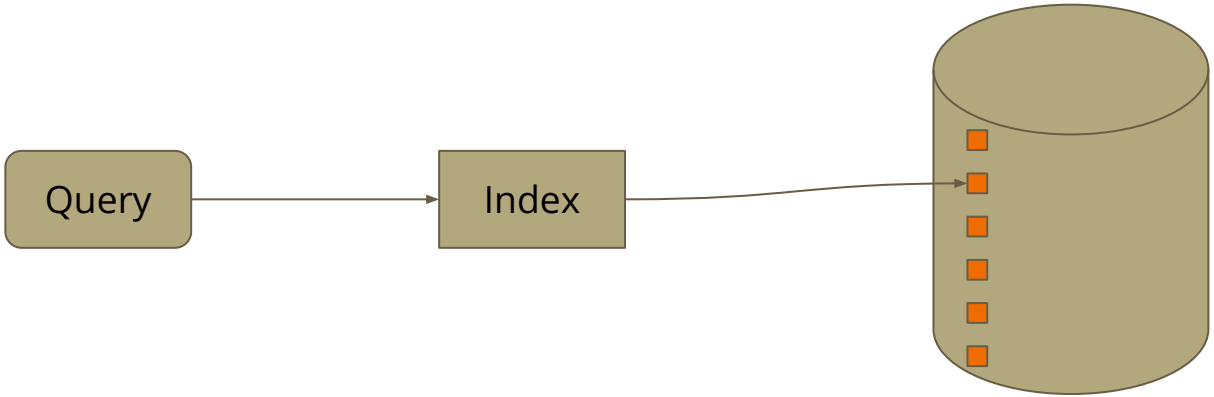
Transformers do not work!!

- Irregular block access



Transformers do not work!!

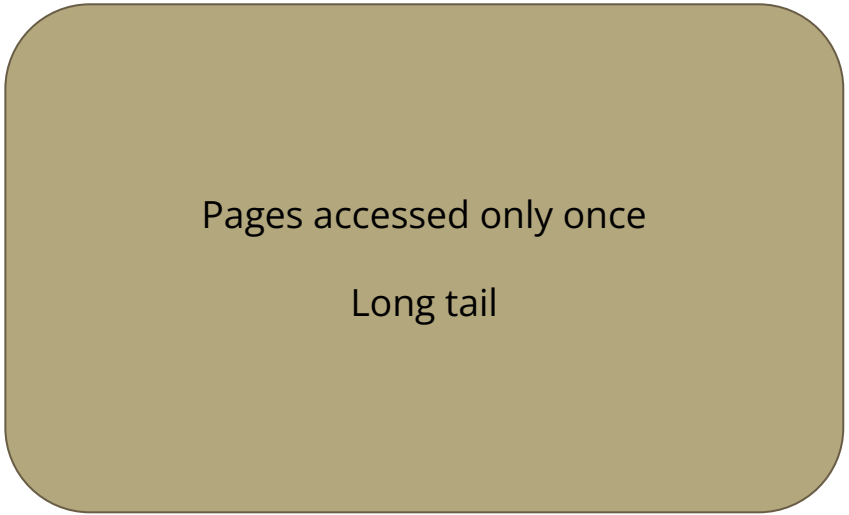
- Irregular block access



Transformers do not work!!

- Page access distribution

Pages accessed many times

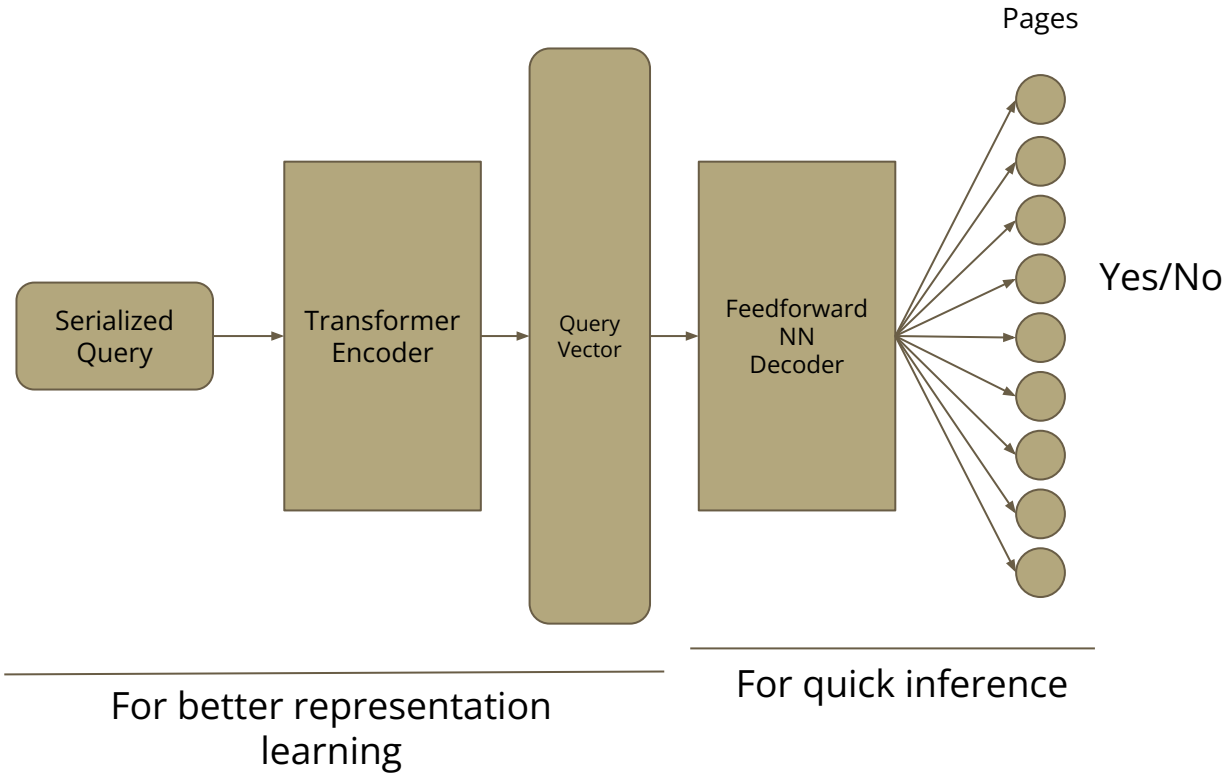


Transformers are bad at learning long tail knowledge

Transformers do not work
NOW

With better and faster models
Prefetch is possible in the future

Hybrid Predictor

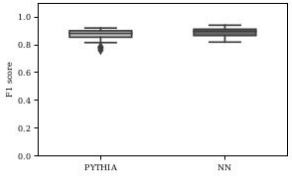


Experiments

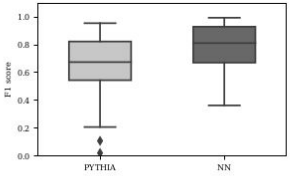
- DSB OLAP benchmark
  - 100GB database
  - Query templates as workloads
- Postgres
  - Enhance postgres to prefetch pages
- Performance metrics
  - F1-score for model accuracy
  - Speedup for prefetch improvement
- Baselines
  - Oracle (ORCL) - knows all future page accesses for a query
  - Nearest neighbour (NN) - knows closest set of page accesses from the training pool

Experiments

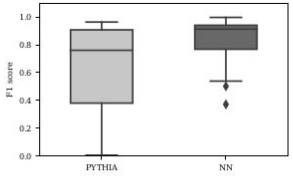
F1-score



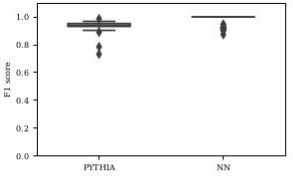
(a) Template 13



(b) Template 18

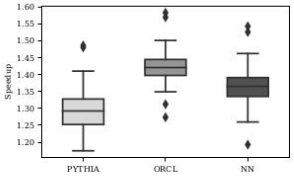


(c) Template 19

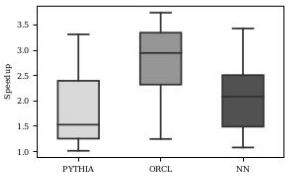


(d) Template 91

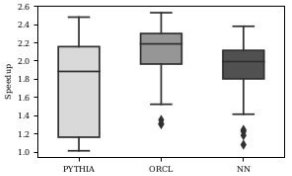
Speedup



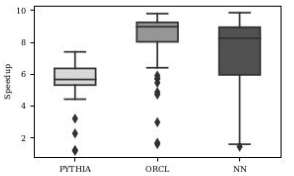
(a) Template 13



(b) Template 18

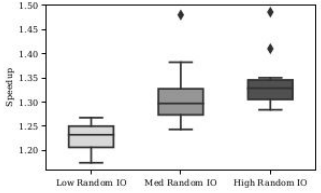


(c) Template 19

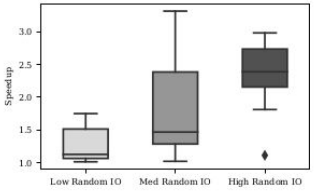


(d) Template 91

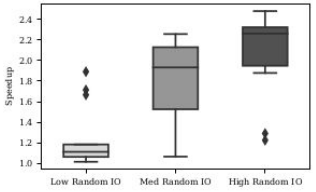
Impact of random reads



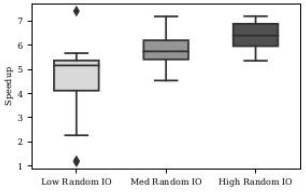
(a) Template 13



(b) Template 18

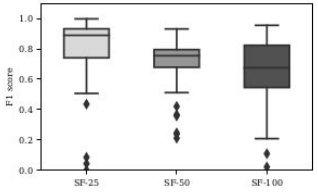


(c) Template 19

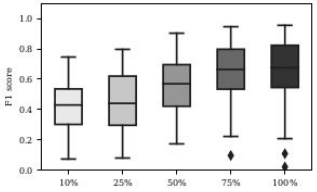


(d) Template 91

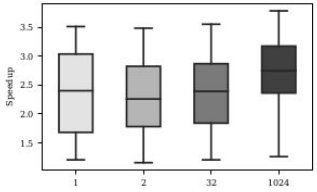
Impact of miscellaneous factors on template-18



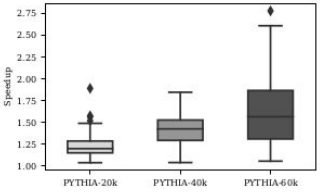
(a) Scale Factor



(b) Training Data Size

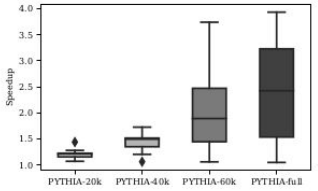


(c) Readahead Size Window

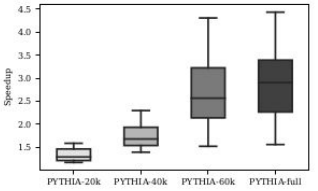


(d) Speedup when predicting top-k frequent pages

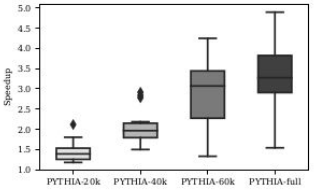
Impact of concurrent queries on template-18



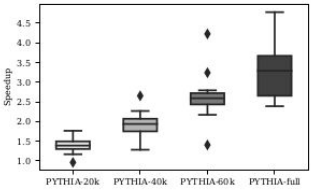
(a) 2 concurrent queries



(b) 3 concurrent queries



(c) 4 concurrent queries



(d) 5 concurrent queries

Conclusion

- Page access patterns are predictable
- Prefetching provides high performance benefits
- Integration with Postgres for experimental evaluation

Questions?