Concurrent Processing of Text Corpus Queries

Radoslav Rábara, Pavel Rychlý

RASLAN 2015-12-04

This work has been partly supported by the Czech Science Foundation under the project GA15-13277S.

Motivation

- text corpora are huge collection of texts – up to billions of words
- queries evaluation can be slow

Manatee

- corpus manager
- implemented in C++
- FastStream and RangeStream

Go (golang)

- new programming language
- build-in concurrency primitives:
  - goroutine
  - channel

Implementation

- written in Go
- FastChan and RangeChan
- uses goroutines and channels

Implementation – data exchange problem

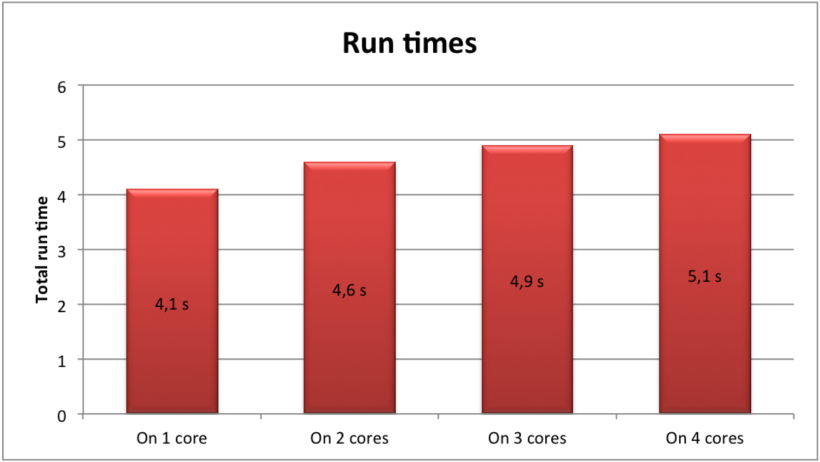

Figure : More cores caused worse performance.

Implementation – data exchange solution

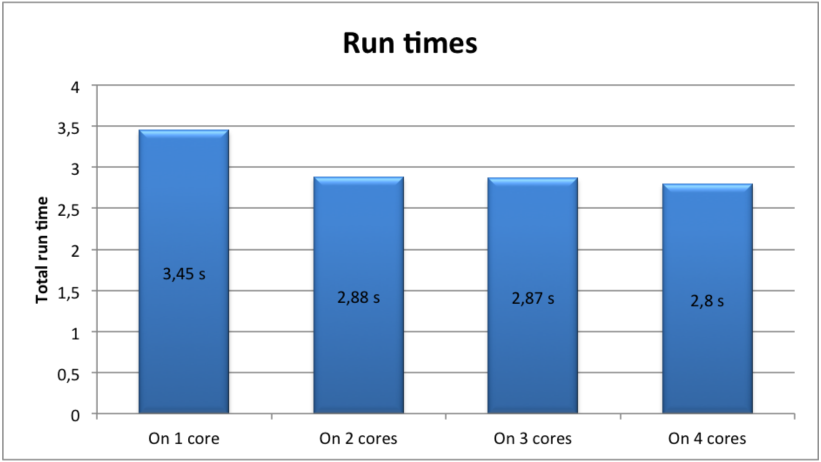

Figure : Sending positions as batches improves performance.

Performance evaluation

- simple benchmark
- complex and quite extreme corpus queries

Performance evaluation

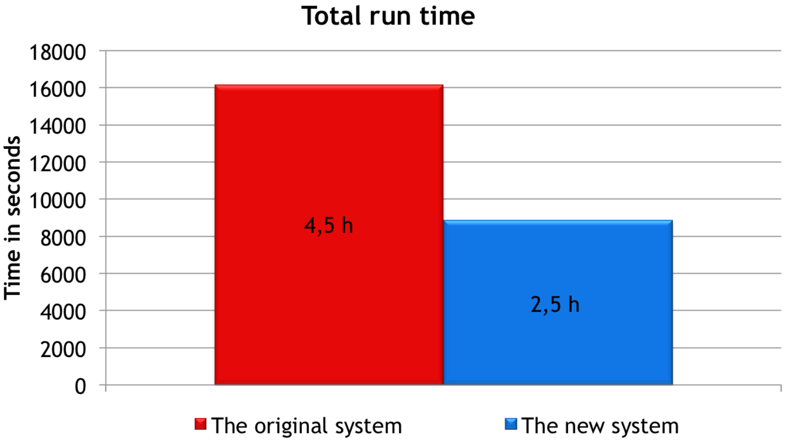

Figure : Compared time of the queries evaluation between the original and new implementation.

Performance evaluation - scalability of the new system

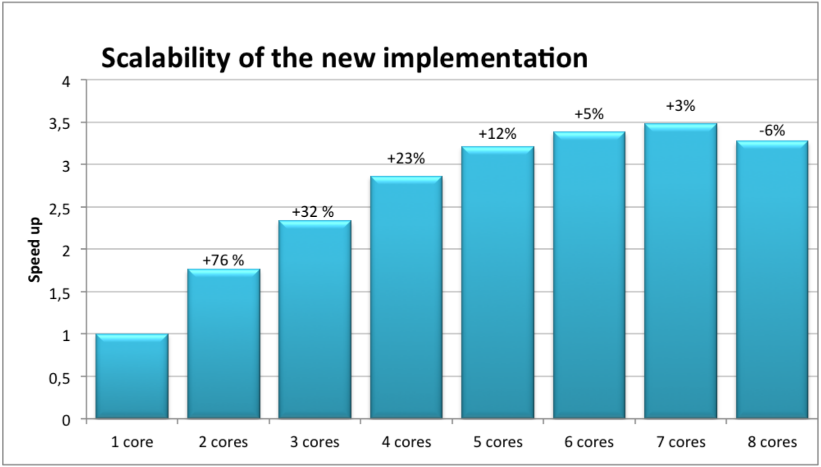

Figure : Scalability of the new implementation.

Comparing lines of code

- the original implementation
  - C++
  - iterators
- the new implementation
  - Go
  - goroutines and channels
  - does not implement all the functionality

Comparing lines of code

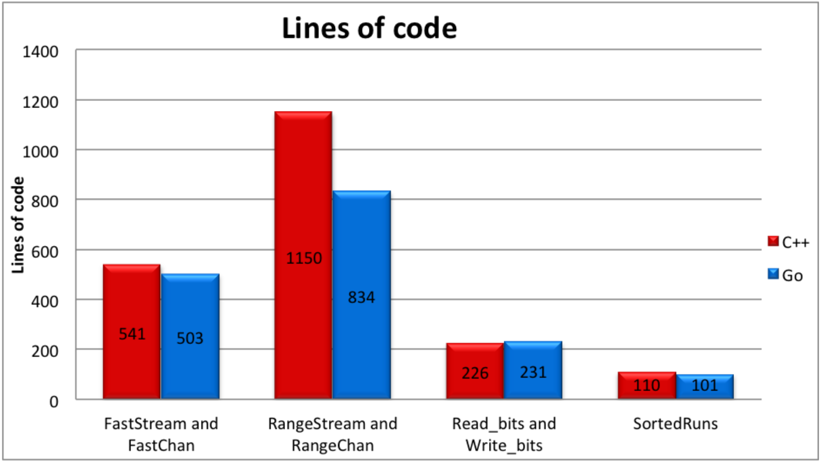

Figure : Graph comparing lines of code of the original and new implementation

Conclusion

The new system has:

- better performance
- shorter source code

Therefore, it will replace the original system.

Conclusion

The new system has:

- better performance
- shorter source code

Therefore, it will replace the original system.