UNIVERSITÀ DEGLI STUDI FIRENZE

Ngrams

Sequential and parallel implementations of bigrams and trigrams in Java

Armand Palla-7042287

UNIVERSITÀ DEGLI STUDI FIRENZE

Ngrams

- The main goal is to compute and estimate occurrences of bigrams and trigrams in a certain text. More in general:
- • A sequence of two letters (e.g. of) is called a bigram.
- • A three-letter sequence (e.g. off) is called a trigram.
- • The general term n-gram means ‘sequence of length n’.

Libraries that we have used:

```
import java.nio.file.Paths;
import java.nio.file.Files;
import java.util.ArrayList;
import java.util.concurrent.*;
import java.util.stream.Collectors;
import java.util.stream.Stream;
import java.util.concurrent.ConcurrentHashMap;
```

Languages

- We have used the Java language for both versions:

Sequential

- n: Number of grams
- file: contains the

  characters from text file

```
Data: n, file
1.  for i = 0 to file.length-n+1 do
2.      key = "";
3.      for j = 0 to n - 1 do
4.          key = key + file[i+j];
5.      end
6.  end
```

Data Structure for sequential version

- HashMaps are the selected data structure to store bigrams and trigrams.
- A HashMap store items in “key/value” pairs and we can accesss them by an index of another type (eg. a String).
- It can’t be shared between many threads without proper synchronization code.
- Hashmaps make no guarantees as to the order of the map.

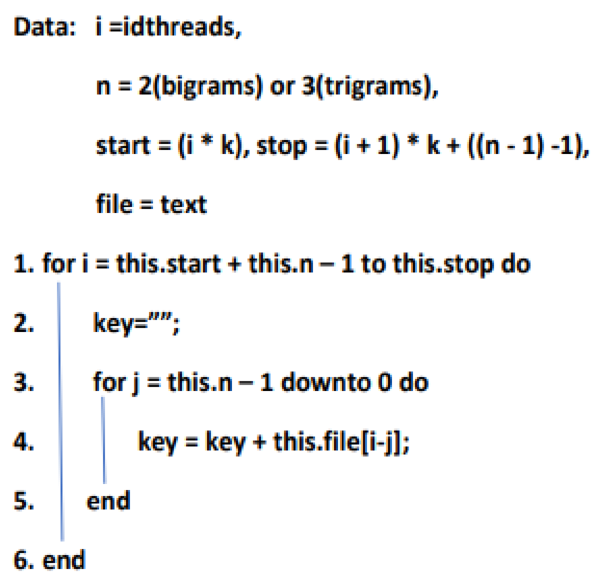
Parallel version

Parallel

- id = idthread
- k = floor(text.length/nThreads)
- n = ngrams • start = (k * i)
- stop = (i+1)*k + ((n-1)-1)
- files = text

Thread’s attributes

- id
- start
- stop
- file



```
Data:  i =idthreads,
       n = 2(bigrams) or 3(trigrams),
       start = (i * k), stop = (i + 1) * k + ((n - 1) -1),
       file = text
1. for i = this.start + this.n – 1 to this.stop do
2.     key="";
3.     for j = this.n – 1 downto 0 do
4.         key = key + this.file[i-j];
5.     end
6. end
```

Parallel implementation

- Idea:

i. Divide the text in as many parts as are the thread instances.

ii. Make the search of bigrams or trigrams on a single part to a single thread.

iii. Use the **k=floor(fileLen/realThreads)** as the dimension of text for each thread in order to separate the text in as many parts as are threads.

Parallel implementation - Java thread

- • Declare a thread class which implements callable.
- • Implement the call() method which computes bigrams and trigrams as described before.
- • Implement a HashMerge() function to merge the maps returned from threads.
- • Instantiate a Future array and an ExecutorService specifying the thread pool size.
- • Use the ExecutorService object to submit the compute method and get the results through .get() Future method.

Data Structure of parallel implementation

- We have used ConcurrentHashMap because it allows concurrent modifications of the Map from several threads without the need to block them.
- ConcurrentHashMap class is thread-safe, multiple threads can operate on a single object without any compilations.
- The object is divided into a number of segments according to the concurrency level.

Results:

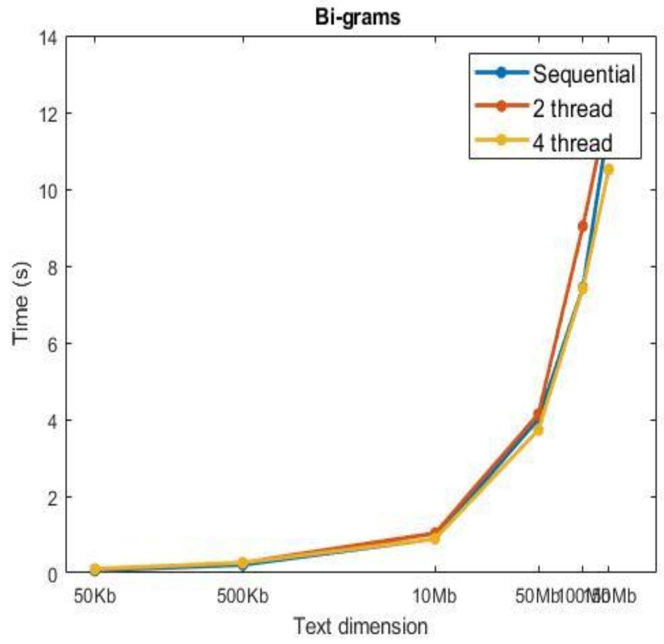
- In the tests of the application, we have studied the behavior of SpeedUp.

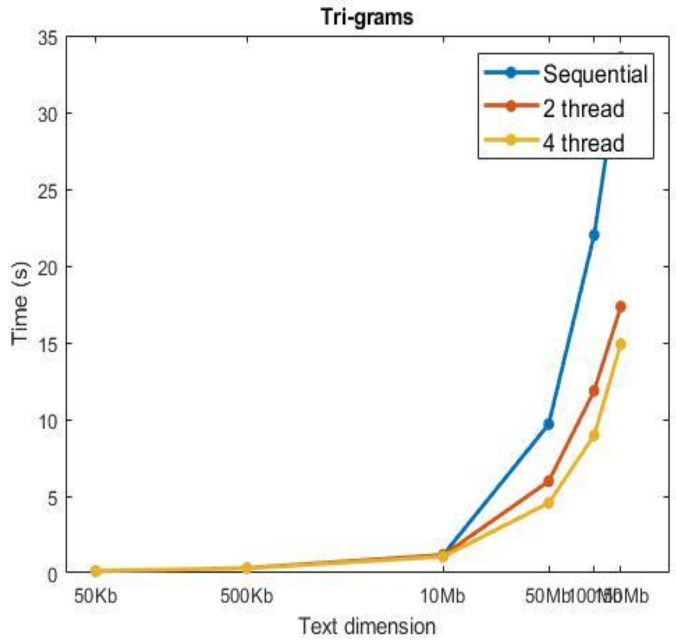
Sp = ts / tp

- The results of this program depends on two main things:
- • Number of threads: 2 or 4
- • Size of files: 50KB, 500KB, 10MB, 50MB, 100MB,150MB

UNIVERSITÀ DEGLI STUDI FIRENZE

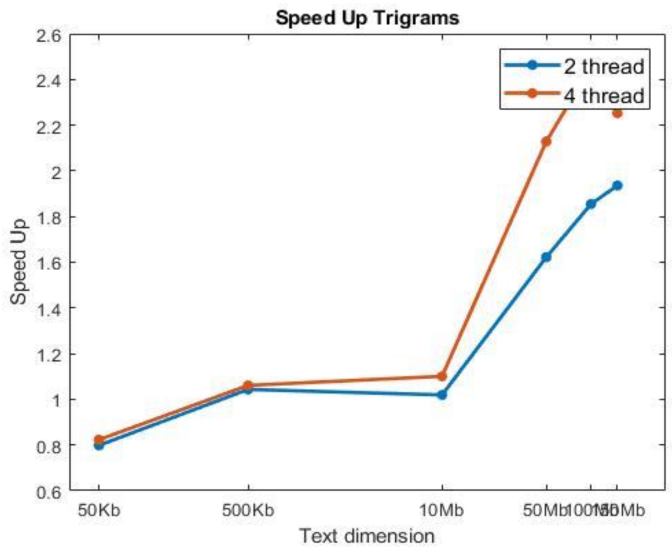
Results:

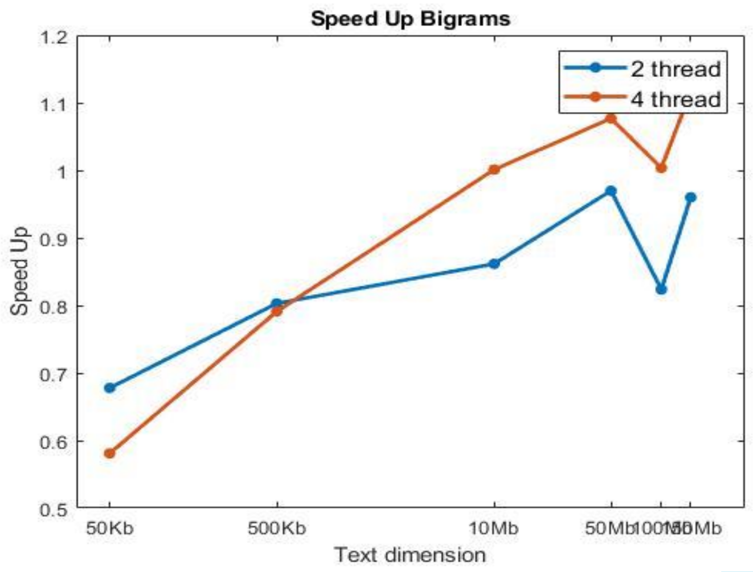




UNIVERSITÀ DEGLI STUDI FIRENZE

Results:


Speed Up Trigrams
2 thread
4 thread
Speed Up
Text dimension
50Kb
500Kb
10Mb
50Mb
100Mb
150Mb
0.6
0.8
1
1.2
1.4
1.6
1.8
2
2.2
2.4
2.6


Speed Up Bigrams
2 thread
4 thread
Speed Up
Text dimension
50Kb
500Kb
10Mb
50Mb
100Mb
150Mb
0.5
0.6
0.7
0.8
0.9
1
1.1
1.2

UNIVERSITÀ DEGLI STUDI FIRENZE

Conclusions:

- For smaller size of texts the speed up in bigrams is more higher with 2 threads rather than 4 threads.
- For bigger size of texts the speed up in bigrams is more higher with 4 threads rather than 2 threads.
- For trigrams the speed up for smaller size of texts, related to threads is approximately the same .
- For larger size of texts the speed up is more higher with 4 threads rather than threads.

UNIVERSITÀ DEGLI STUDI FIRENZE