

# Parallellisme - Project report

Hans Van der Ougstraete

July 14, 2020

# Contents

<b>1</b>	<b>Implementatie</b>	<b>3</b>
1.1	Fase 1 . . . . .	3
1.2	Fase 2 . . . . .	3
<b>2</b>	<b>Evaluatie</b>	<b>3</b>
2.1	Fase 1 . . . . .	3
	2.1.1 Overhead . . . . .	3
	2.1.2 Speed-up . . . . .	5
2.2	Fase 2 . . . . .	5

# 1 Implementatie

## 1.1 Fase 1

Fase 1 is opgedeeld in twee delen, het eerste deel maakt een lijst met relevante businesses door in de base case de evaluaterevalence methode uit SequentialSearch aan te roepen. De Yelpdata wordt dus in stukken opgedeeld tot het kleinste deel een business is. Deze lijst wordt in het tweede deel aan de hand van een parallele mergesort gesorteerd. Deze code is terug te vinden in de ParallelSearch klasse.

## 1.2 Fase 2

In progress

# 2 Evaluatie

De benchmarks op eigen computer (quad core i5-2300 @2.80GHz) zijn 4 maal uitgevoerd. Voor SequentialSearch, ParallelSearch met 1, 2 workers en ParallelSearch met 4 workers. Dit telkens 1500 keer. De resultaten worden per preset weergegeven.

## 2.1 Fase 1

Runtime ms	Preset1	Preset2	Preset3
Parallel P=1	466	514	145
Parallel P=2	304	328	111
Parallel P=4	273	291	108
Sequential	451	381	134

Table 1: Gemiddelde runtime in miliseconden

### 2.1.1 Overhead

Overhead	Preset1	Preset2	Preset3
T1/Tseq	1.03	1.35	1.08

Table 2: Overhead waarden voor de 3 presets.

Bovenstaande tabel toont de overhead, deze overhead van het verdelen in threads is niet aanwezig bij de sequentiële versie omdat de sequentiële versie niet opdeelt in alsmal kleinere taken. De overhead is verschillend per preset omdat het opdelen van het werk afhangt van de respectievelijke data. De overhead kan

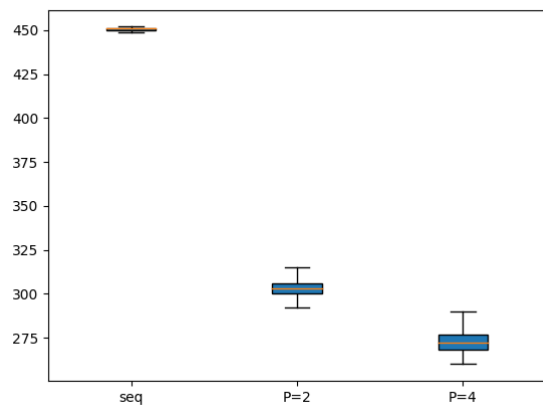


Figure 1: Preset1

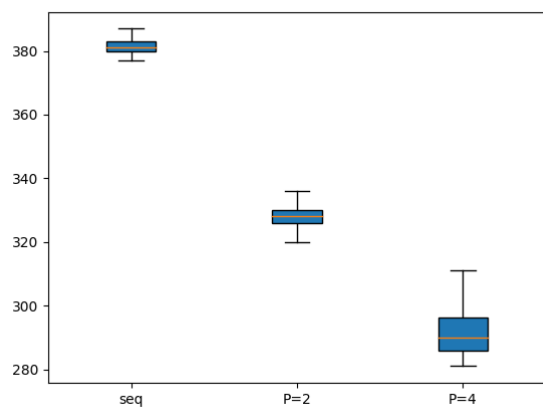


Figure 2: Preset2

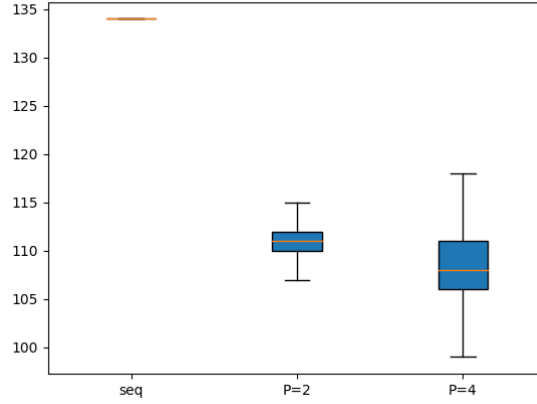


Figure 3: Preset3

kleiner gemaakt worden door een goede cut-off waarde te gebruiken, hierdoor wordt er sneller sequentieel gewerkt en daardoor minder opgedeeld in subtaken.

### 2.1.2 Speed-up

Speed-up	Preset1	Preset2	Preset3
$T_{seq}/T_2$	1.48	1.16	1.21
$T_{seq}/T_4$	1.65	1.31	1.24

Table 3: Speed-up values for all 3 presets.

Een perfecte speed-up zou gelijk zijn aan het aantal extra processors, dit is hier niet het geval. De speed-up is ook verschillend per preset, het aantal businesses en reviews beïnvloeden dit volgens mij. Bij preset 3 is er nauwelijks verschil tussen  $T=2$  en  $T=4$ , ik veronderstel dat dit is omdat de data klein is. het meeste winst wordt geboekt bij preset 1, de preset met het meest aantal businesses.

## 2.2 Fase 2