Comparison of sequential and parallel algorithms for word and context count

Names: Eduardo Ferreira, Francieli Zanon, Aline Villavicencio
Groups: Processamento de linguagem natural e
Processamento paralelo e distribuido (UFRGS)

Motivation

Parallelize one of the steps for Distributional Thesaurus creation

Create faster Distributional Thesaurus

Used in many NLP applications

- Machine Translation
- Question Answering

Needs great amount of data to be built

Agenda

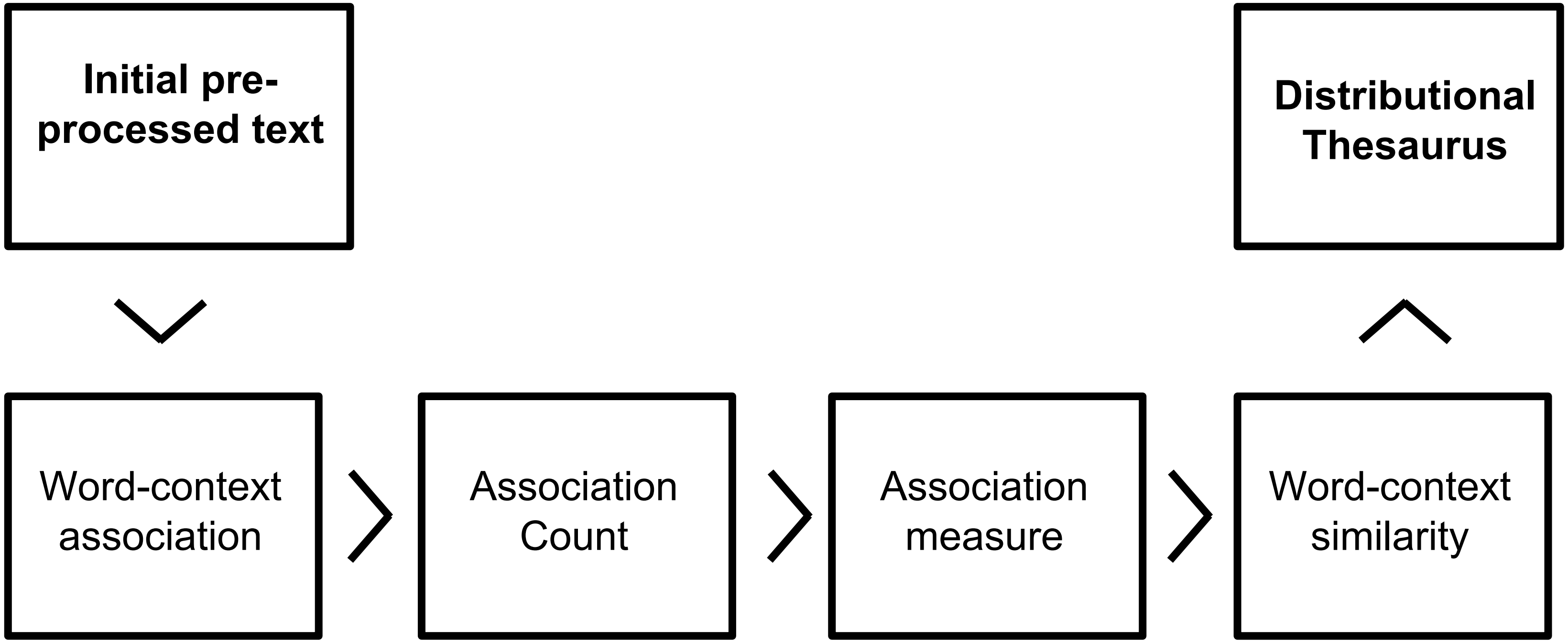
- Distributional Thesaurus Creation
- Parallel Version
- Results

Distributional Thesaurus Creation

A thesaurus is a list of words associated by a specific characteristic.

word	**synonyms**
abandon	**leave, desert, give up, surrender, ...**
abide	**tolerate, accept, endure, stand, ...**

Distributional Thesaurus Creation



Initial pre-processed text

Word-context association

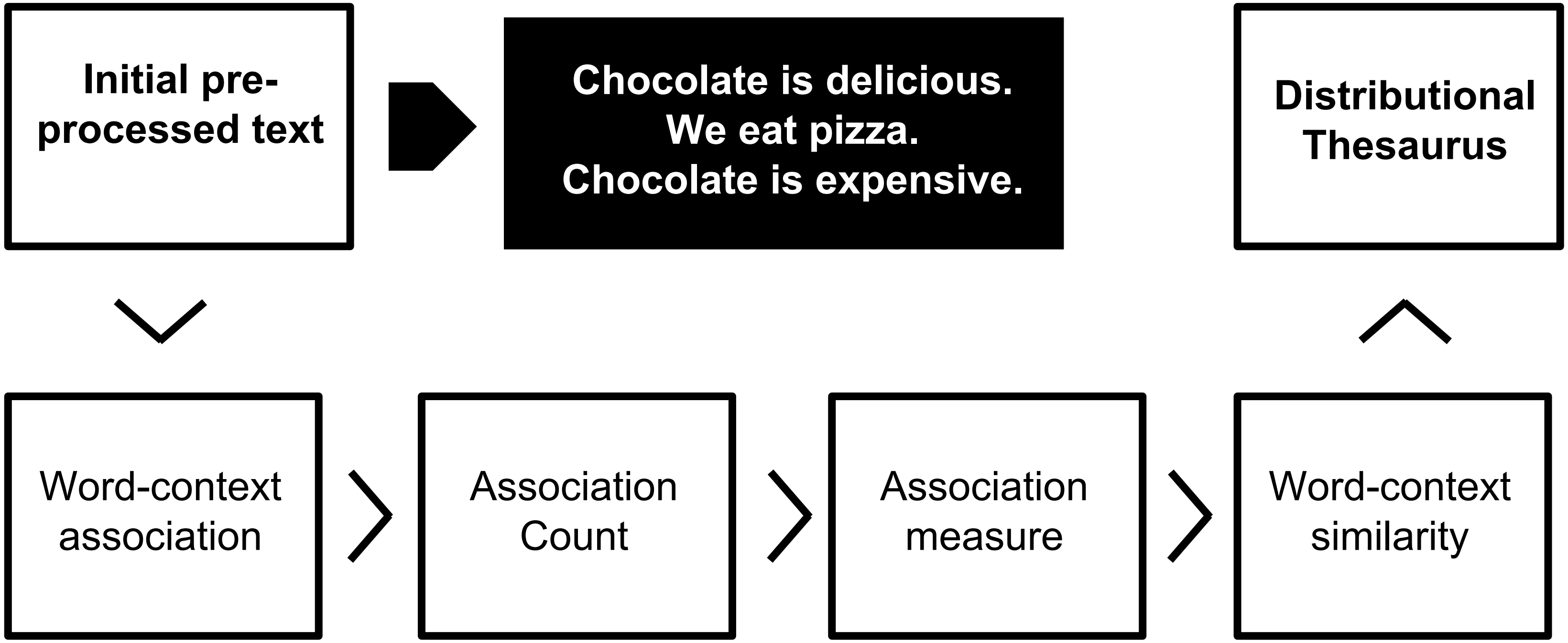
Association Count

Association measure

Word-context similarity

Distributional Thesaurus

Distributional Thesaurus Creation



Initial pre-processed text

Chocolate is delicious.
We eat pizza.
Chocolate is expensive.

Distributional Thesaurus

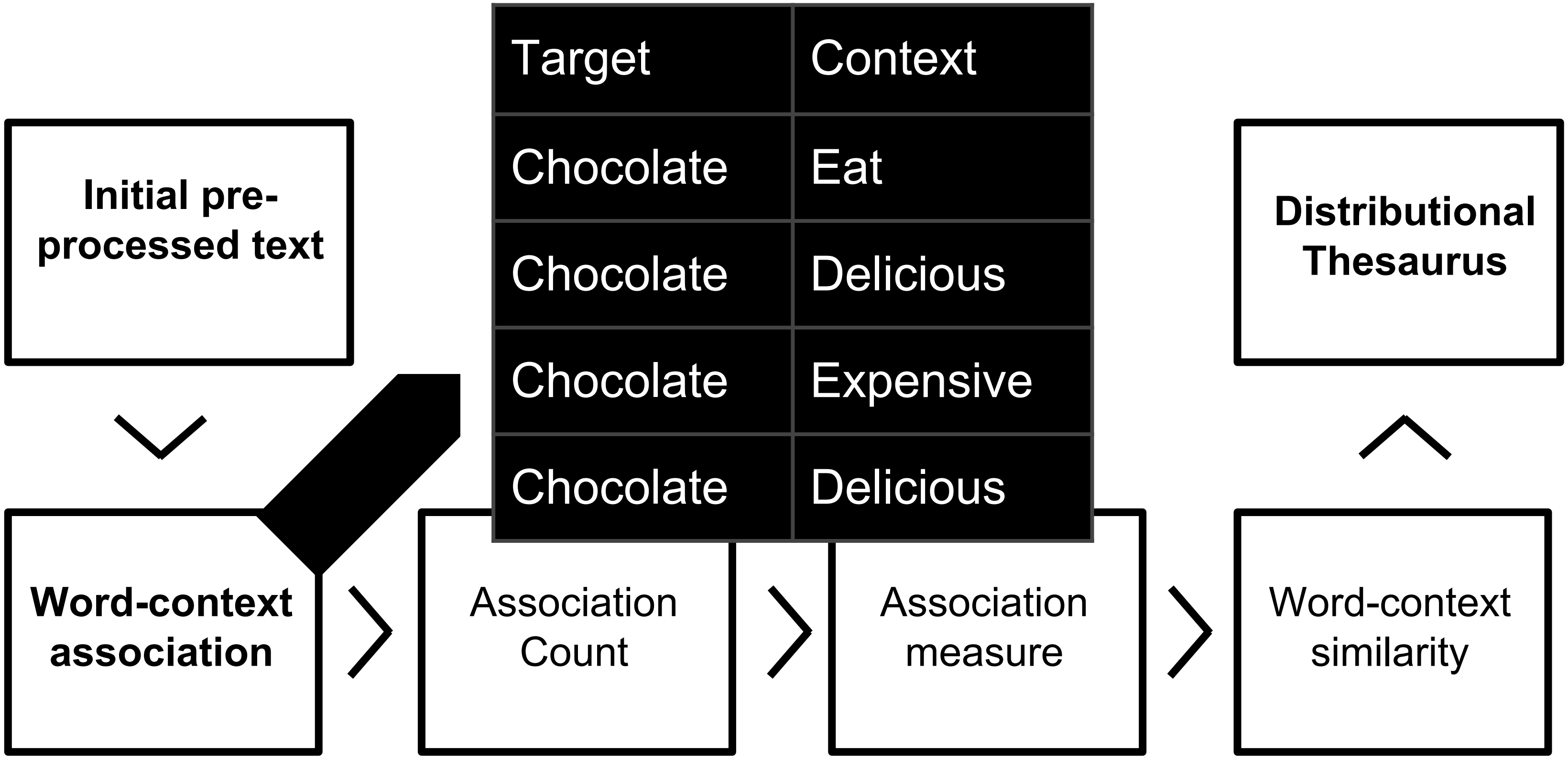
Word-context association

Association Count

Association measure

Word-context similarity

Distributional Thesaurus Creation



Initial pre-processed text

Word-context association

Association Count

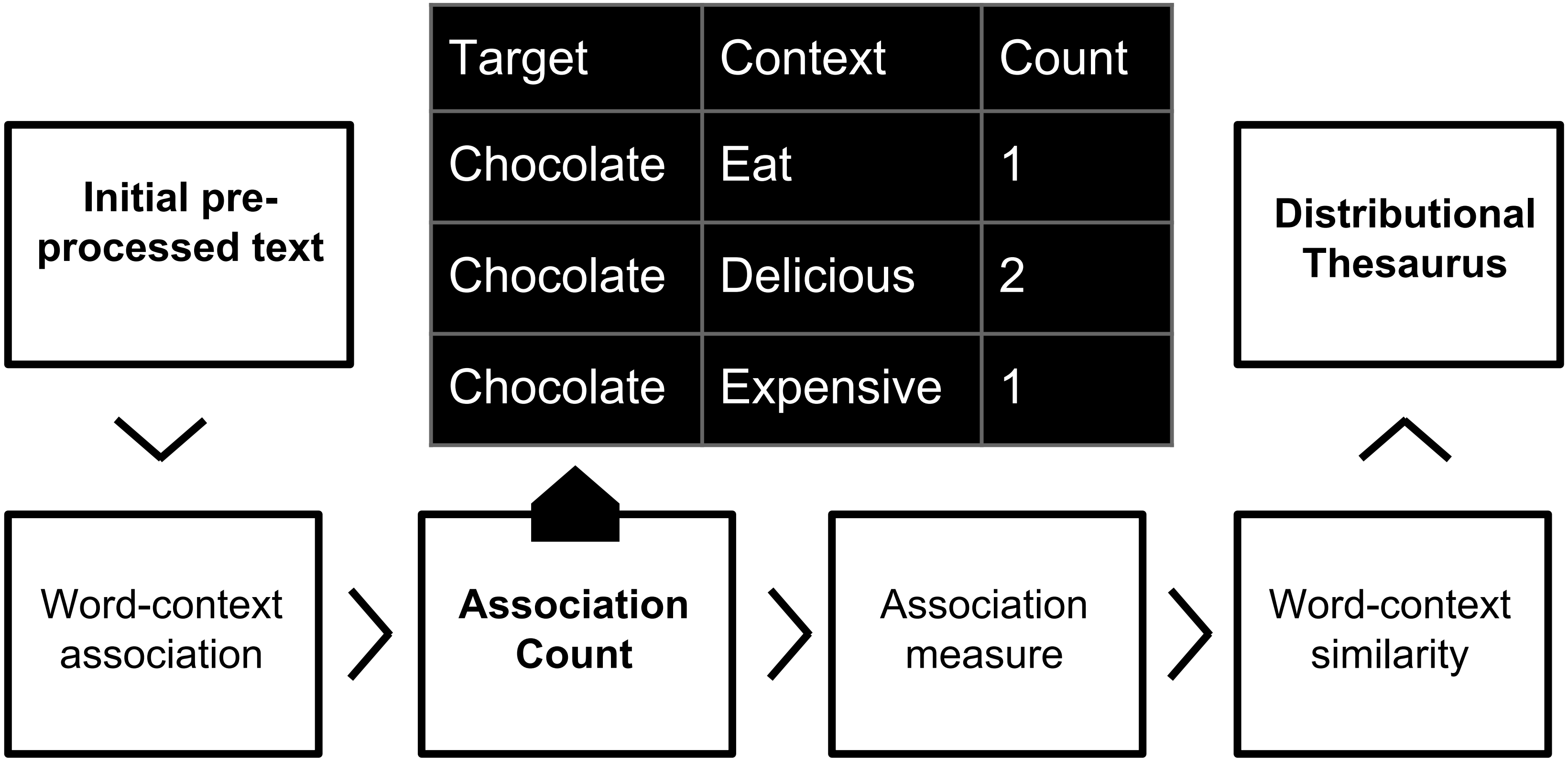
Association measure

Word-context similarity

Distributional Thesaurus

Target	Context
Chocolate	Eat
Chocolate	Delicious
Chocolate	Expensive
Chocolate	Delicious

Distributional Thesaurus Creation



Initial pre-processed text

Word-context association

Association Count

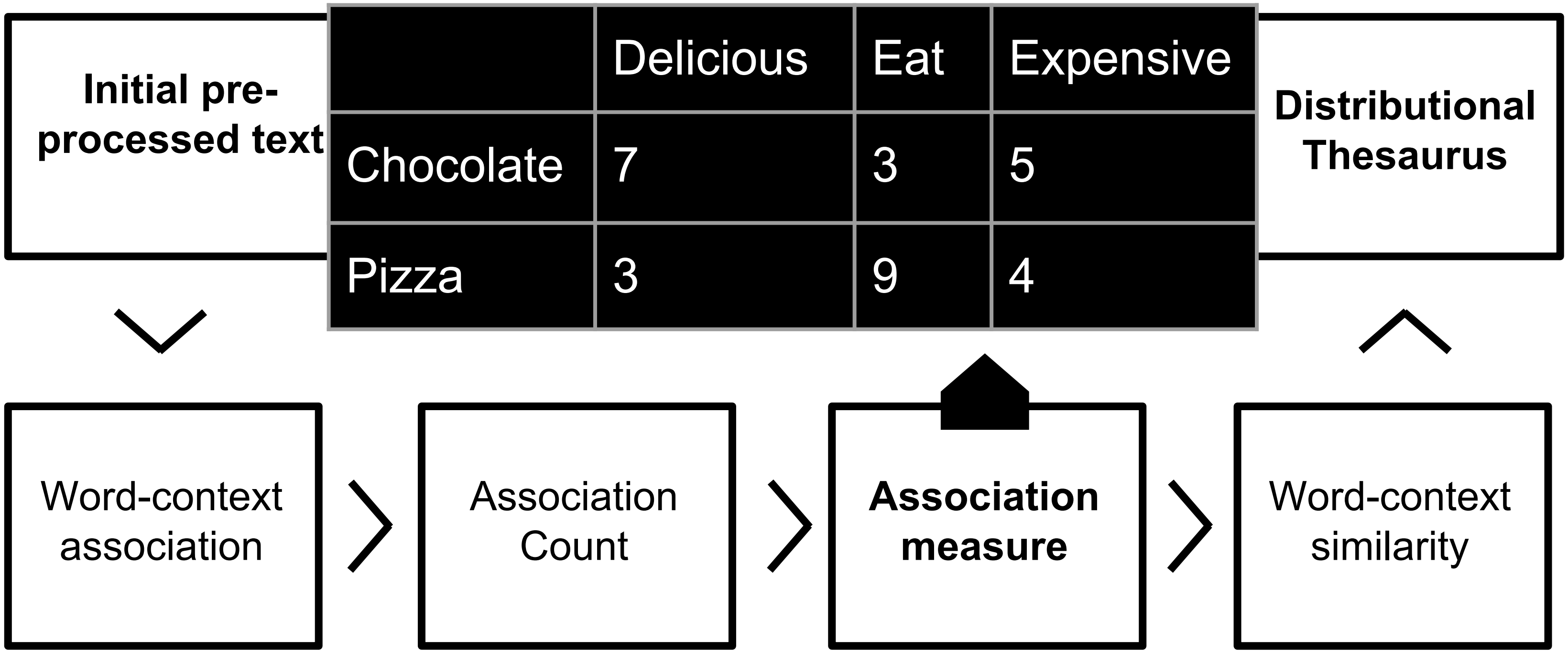
Target	Context	Count
Chocolate	Eat	1
Chocolate	Delicious	2
Chocolate	Expensive	1

Association measure

Word-context similarity

Distributional Thesaurus

Distributional Thesaurus Creation



Initial pre-processed text

Distributional Thesaurus

Word-context association

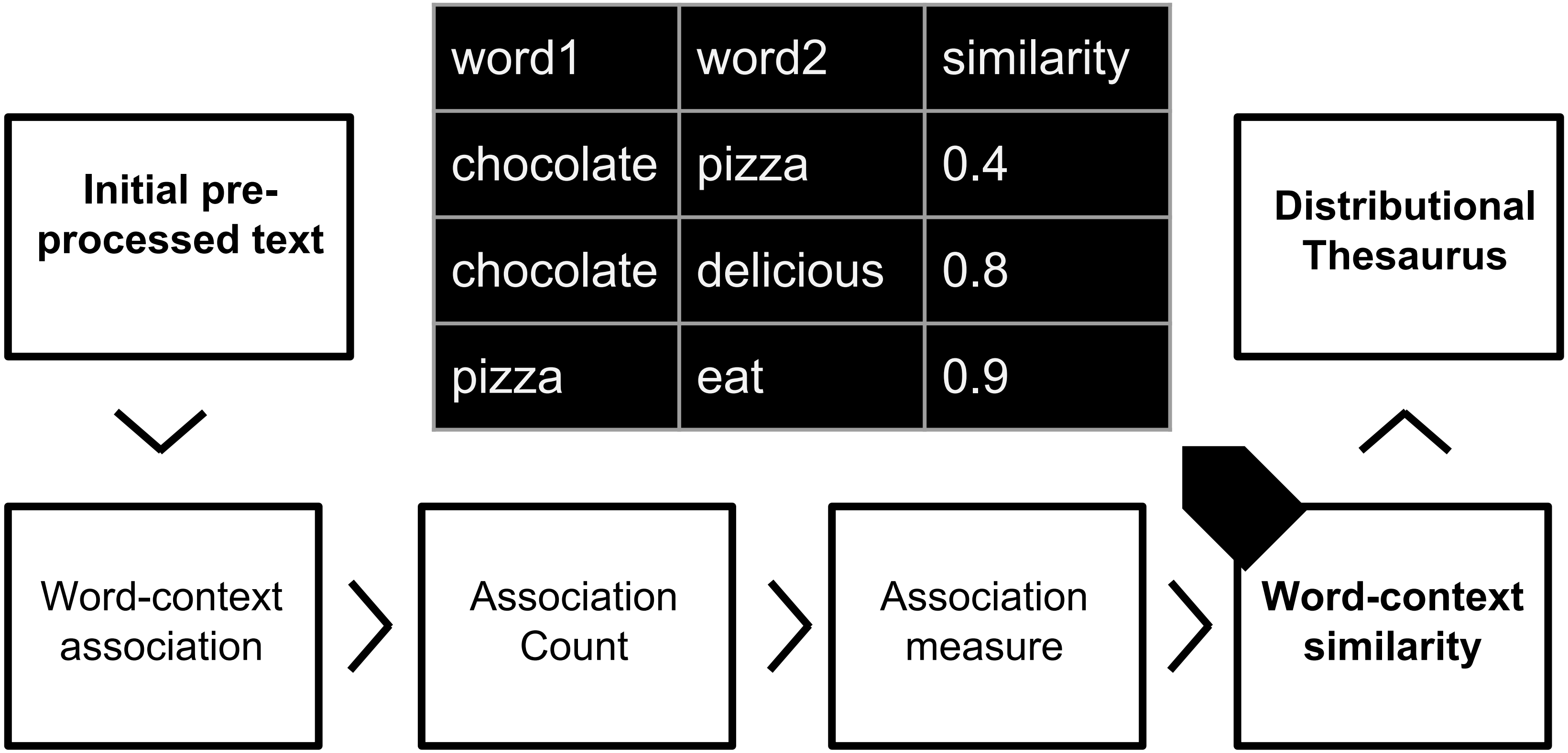
Association Count

Association measure

Word-context similarity

	Delicious	Eat	Expensive
Chocolate	7	3	5
Pizza	3	9	4

Distributional Thesaurus Creation



word1	word2	similarity
chocolate	pizza	0.4
chocolate	delicious	0.8
pizza	eat	0.9

Initial pre-processed text
Word-context association
Association Count
Association measure
Word-context similarity
Distributional Thesaurus

Agenda

- Distributional Thesaurus Creation
- Parallel Version
- Results

Parallel version

- Sequential process is too slow
- Fits the MapReduce paradigm
  - Map: input text divided in multiple parts
  - Reduce: results are grouped together

Parallel version

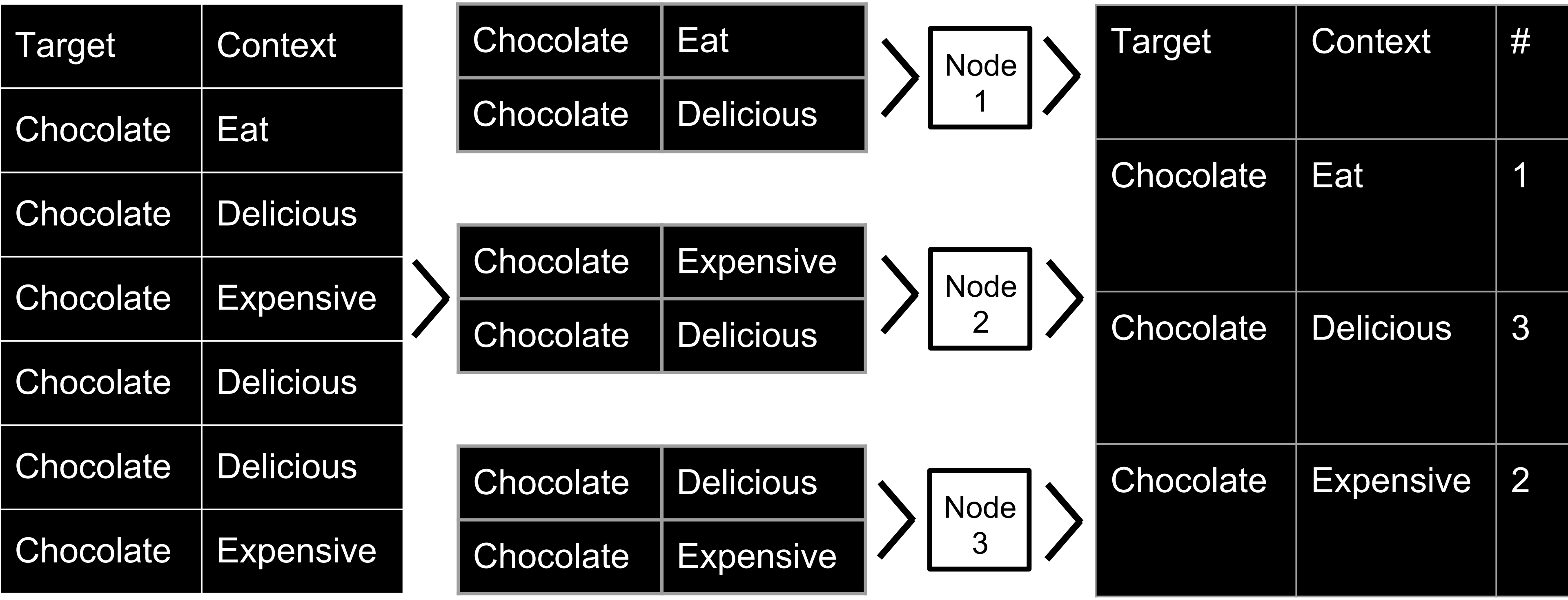
Spark framework

Scala

Tests executed in Sagitaire cluster Grid 5000

up to 40 nodes used, each one with 2 cores.

Parallel version



Target	Context
Chocolate	Eat
Chocolate	Delicious
Chocolate	Expensive
Chocolate	Delicious
Chocolate	Delicious
Chocolate	Expensive

Chocolate	Eat
Chocolate	Delicious

Node 1

Chocolate	Expensive
Chocolate	Delicious

Node 2

Chocolate	Delicious
Chocolate	Expensive

Node 3

Target	Context	#
Chocolate	Eat	1
Chocolate	Delicious	3
Chocolate	Expensive	2

Agenda

- Distributional Thesaurus Creation
- Parallel Version
- Results

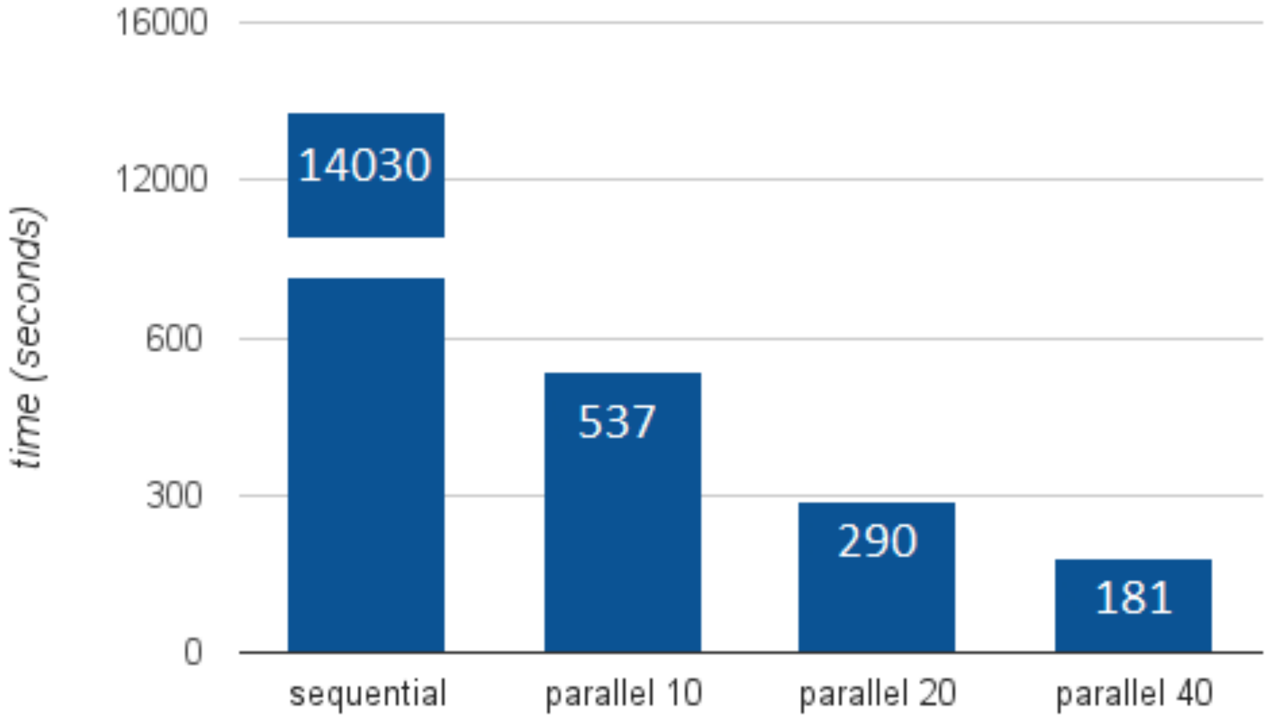
Results

	68 KB	
	sequential	parallel 40
time (in s)	0.09	45.31
speedup		0.0019
eficiency		0.000024

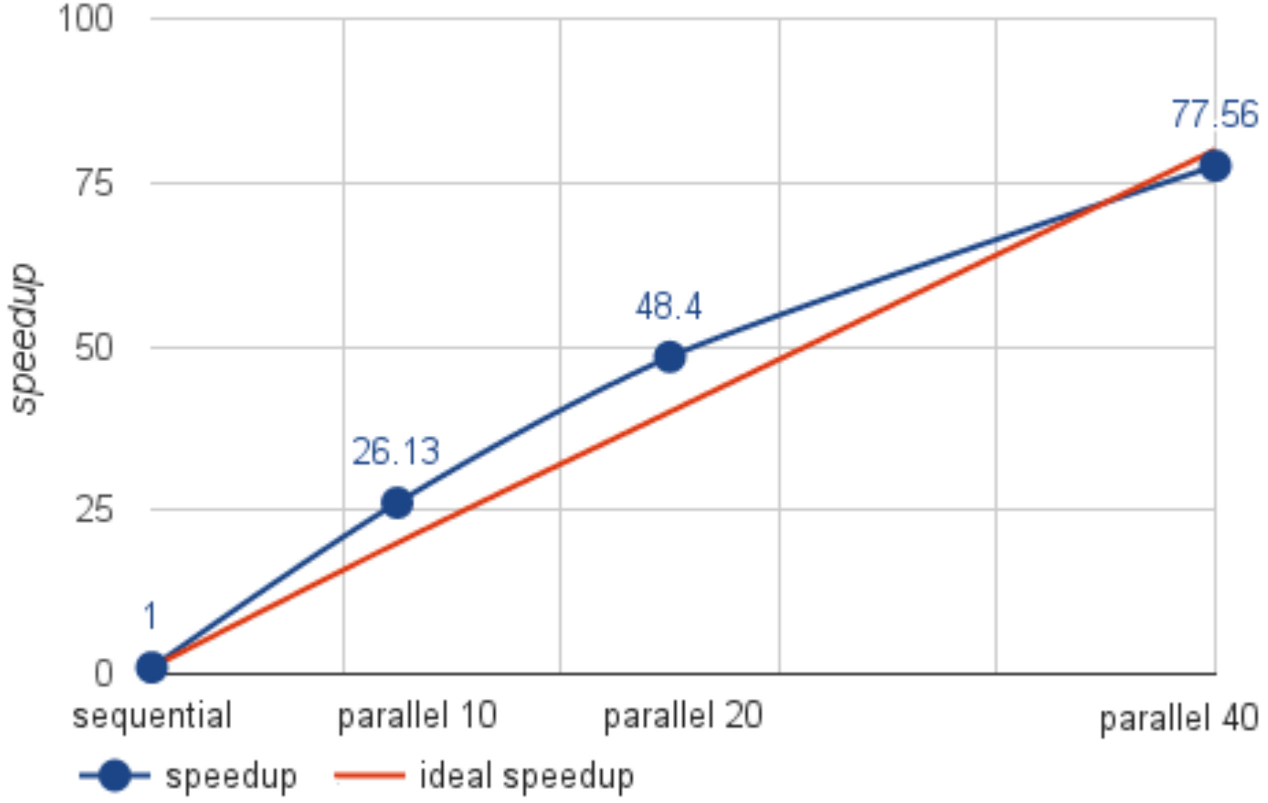
Results

	11 GB			
	sequential	parallel 10	parallel 20	parallel 40
time (in s)	14029.8	536.74	289.85	180.87
Std Deviation		1.056	1.46	3.3
speedup		26.13	48.40	77.56
eficiency		1.30	1.21	0.97

Results


16000
12000
600
300
0
time (seconds)
14030
537
290
181
sequential
parallel 10
parallel 20
parallel 40

Results


100
75
50
25
0
speedup
77.56
48.4
26.13
1
sequential
parallel 10
parallel 20
parallel 40
speedup
ideal speedup

Results

	11 GB		
	parallel 10	parallel 20	parallel 40
time (in s)	1466.34	1499.45	1670.47
speedup	9.56	9.35	8.39
eficiency	0.47	0.23	0.10

Conclusions

The goal of this work was to parallelize the word-context count.

Spark reduced significantly the time required for getting word-context counts.

Performance improvement for large corpora.

Future Work

Test the parallelization using other forms of file distribution (HDFS).

Integrate tuple counts with the other 2 steps:

- Association measure
- Word-context similarity