



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Combining Unsupervised and Supervised Parser



Martin Riedl, Irina Alles and Chris Biemann

Language Technology

Technische Universität Darmstadt, Germany

COLING 2014, Dublin, Ireland, August 26 2014, 16:35-17:00

Motivation

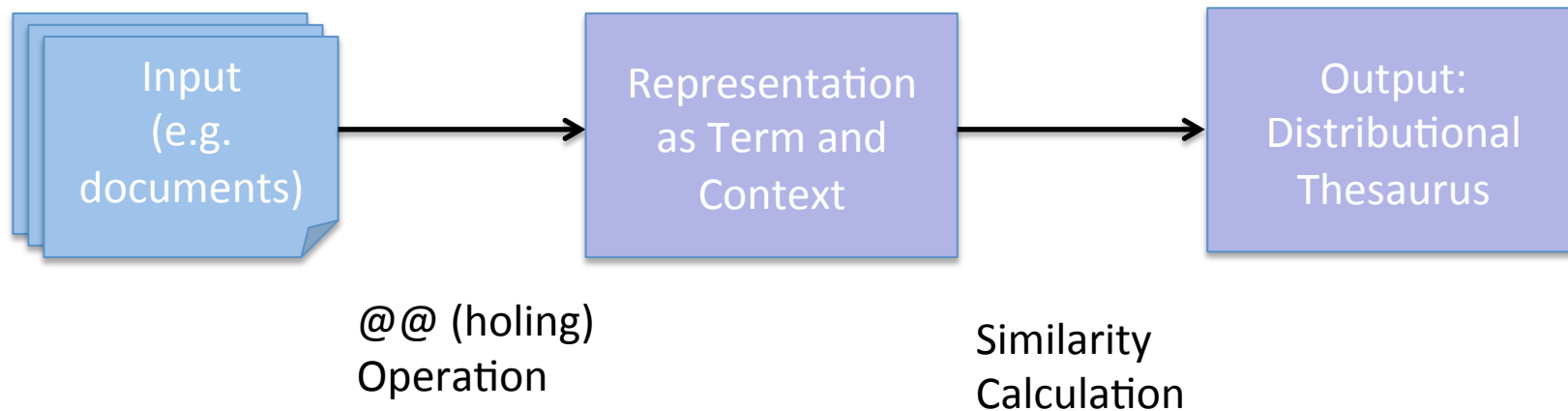
- Dependency parses → Distributional Thesaurus (DT) of high quality
- Unsupervised dependencies → ???
- Combining both → ???

Agenda

- Building Distributional Thesauri (DTs)
- Evaluation of DTs/UPs
- Experimental Setting
- Results
- Conclusion & Outlook

Building a Distributional Thesaurus

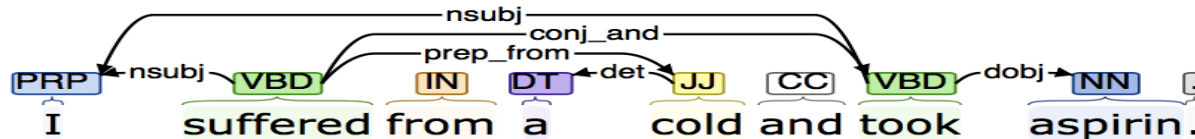
using **JobimText**
Linking Language to Knowledge
with Distributional Semantics



<http://jobimtext.org/>

The @@ operation: JoBim Pairs for Syntax Based Distributional Similarity

SENTENCE:



Dependency Parser:

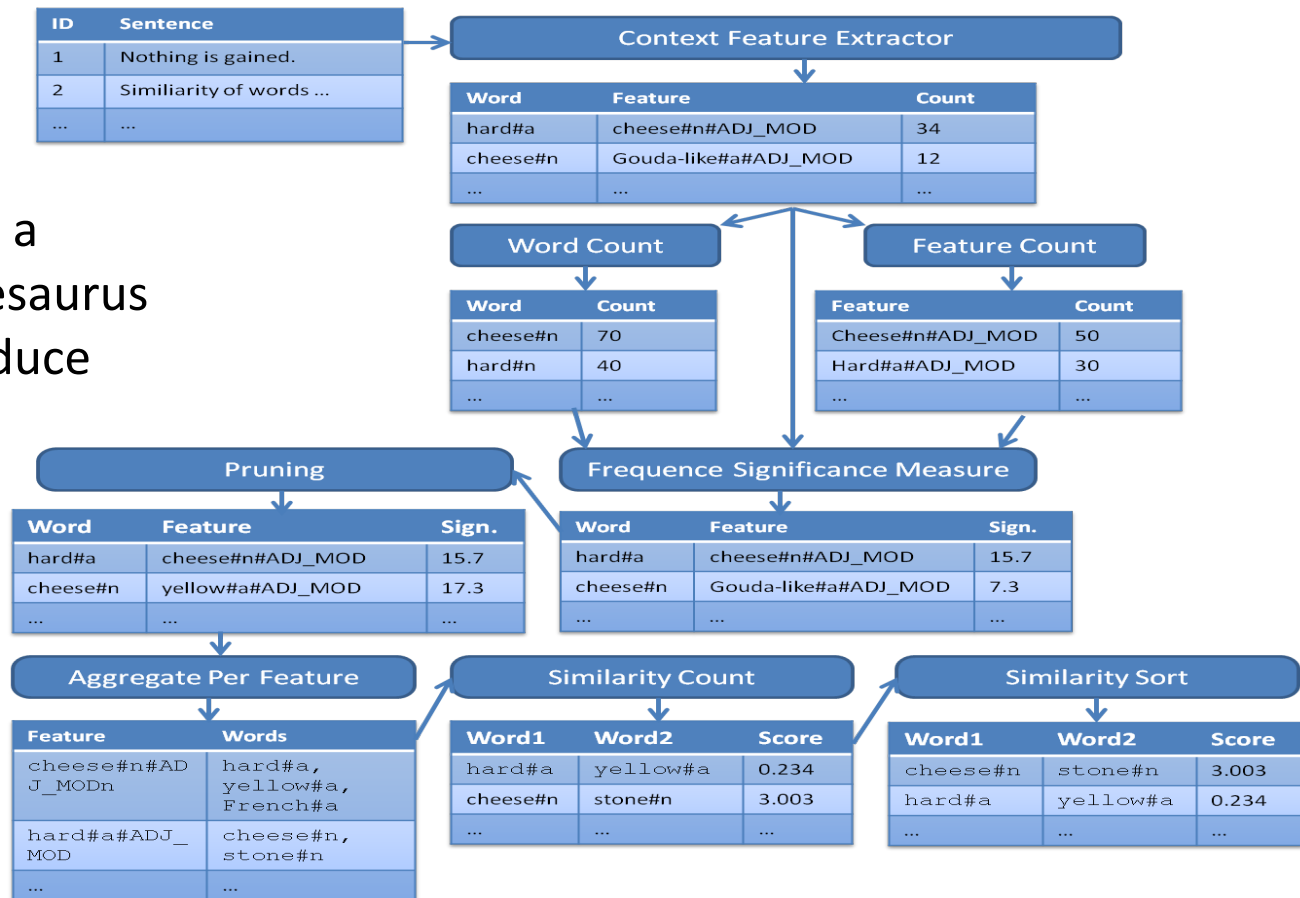
nsubj(suffered, I); nsubj(took, I); root(ROOT, suffered); det(cold, a); prep_from(suffered, cold); conj_and(suffered, took); dobj(took, aspirin)

WORD-dependency PAIRS:

Suffered	nsubj(@@, I)	1
took	nsubj(@@, I)	1
cold	det(@@, a)	1
Suffered	prep_from(@@, cold)	1
Suffered	conj_and(@@, took)	1
took	dobj(@@, aspirin)	1

I	nsubj(suffered, @@)	1
I	nsubj(took, @@)	1
a	det(cold, @@)	1
cold	prep_from(suffered, @@)	1
took	conj_and(suffered, @@)	1
aspirin	dobj(took, @@)	1

Steps to calculate a Distributional Thesaurus (DT) with MapReduce



In our experiments we focus on frequent and rare nouns

Evaluate a DT

Select words from different frequency bands

car
computer
way
...
reinforcement
deployment

Extract top N entries from DT for each word

vehicle	0.33
van	0.50
truck	0.33
jeep	0.50
minivan	0.50
bus	0.50
...	

Compute Path score against (WordNet | GermaNET)

Compute average for all (frequent | rare) words

$\emptyset = 0.220$

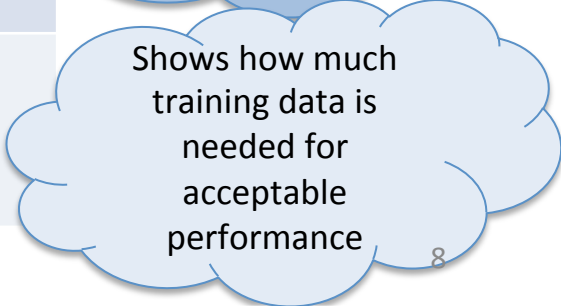
Experimental Setup

- 1) Train UP on Training Corpus
- 2) Apply UP Parser on Test Corpus
- 3) Compute DT with context from UP
- 4) Evaluate DT

Setup	Training Corpus	Test Corpus
Setup A	10k sentences	10k sentences
	100k sentences	100k sentences
	1M sentences	1M sentences
	10M sentences	10M sentences
Setup B	10k sentences	10M sentences
	100k sentences	10M sentences
	1M sentences	10M sentences
	10M sentences	10M sentences



Use Same Training
& Test Corpus



Shows how much
training data is
needed for
acceptable
performance

Baselines & Parsers

	English	German	Use POS
Baseline	Random Parser		no
	Left/Right Branching (Bigram)		no
	Left & Right Branching (Trigram)		no
Supervised	Stanford Parser	Mate Parser	yes
Unsupervised	Gillenwater (method based on DMV)		yes
	UDP (method based on DMV)		yes
	Bisk (EM approach inducing a Combinatory Categorical Grammar)		yes
	Søgaard (Use PageRank and heuristics to connect words)		yes/no
	Seginer (incremental parser using common cover links)		no

Resources

	English	German
Corpus	LCC ¹ English newspaper	LCC ¹ German newspaper
Taxonomy for evaluation	WordNet	GermaNet
words used for evaluation	1000 frequent and 1000 rare nouns	1000 frequent and 1000 rare nouns

¹ <http://corpora.uni-leipzig.de/>

Results English (frequent words): Setup A

		Training (for UP only) and Test Data				
		Parser	10k	100k	1M	10M
Baselines	Random		0.115	0.128	0.145	0.159
	Trigram		0.133	0.179	0.200	0.236
	Bigram		0.140	0.173	0.208	0.246
	Stanford		0.151	0.209	0.261	0.280
Unsupervised Parser	Seginer		0.136	0.176	0.211	0.240
	Gillenwater		0.135	0.159	0.195	0.223
	Søgaard		0.120	0.147	0.185	0.227
	UDP		0.127	0.169	0.204	*
	Bisk		0.118	*	*	*

- Only Seginer can beat the lower baselines on the 1M trained corpus
- Scores increase with more data -> the more the data the better the DT
- UDP did not finish parsing after 157 days, so we skipped it
- Both UP which do not use POS tags lead to the best results

* denotes, that the model could not be computed (errors, time issues)

Results English (frequent words): Setup B

		Training Data (Test is done on 10M)			
		10k	100k	1M	10M
Baselines	Parser				
	Random				0.159
	Trigram				0.236
	Bigram				0.246
	Stanford				0.280
Unsupervised Parser	Seginer	0.200	0.236	0.241	0.240
	Gillenswater	0.220	0.221	0.221	0.223
	Søgaard	0.227	0.227	0.227	0.227
	Bisk	0.220	*	*	*
	UDP	*	*	*	*

- Gillenswater approach can hardly make use of additional training data
- Bisks parser was effectively trained only on 5000 sentences (due to pruning)

* denotes, that the model could not be computed (errors, time issues)

Results English (rare words)

- Results show a similar trend
- Achieve generally lower scores

Results German (frequent words): Setup A

		Training (for UP only) and Test Data			
		10k	100k	1M	10M
Baselines	Parser				
	Random	0.097	0.108	0.123	0.143
	Trigram	0.102	0.130	0.159	0.179
	Bigram	0.112	0.130	0.163	0.192
Unsupervised Parser	Mate	0.111	0.126	0.170	0.204
	Seginer	†0.113	†0.137	0.171	0.208
	Gillenwater	0.104	0.118	0.132	*
	Søgaard	0.104	0.123	0.161	0.193
	UDP	0.107	0.129	0.151	*
	Bisk	0.101	*	*	*

- Seginer outperforms the upper baseline
- Dependency relations from Mate seem to be very sparse
- Søgaard and Seginer achieve good results, when using large data

† significant improvement (paired t-test $p < 0.01$) against the Mate parser ¹⁴
 * denotes, that the model could not be computed (errors, time issues)

Results German (frequent words): Setup B

		Training (for UP only) and Test Data			
		10k	100k	1M	10M
Baselines	Parser				
	Random				0.143
	Trigram				0.179
	Bigram				0.192
	Mate				0.204
Unsupervised Parser	Seginer	0.153	0.186	0.200	0.208
	Gillenwater	0.189	0.190	0.189	*
	Søgaard	0.193	0.193	0.193	0.193
	Bisk	0.185	*	*	*
	UDP	*	*	*	*

- Similar trend as for English

* denotes, that the model could not be computed (errors, time issues)

Combining Thesauri

- We compute the Holing operation
- Combine different feature combinations
- Compute a DT on 10M sentences
 - Our approach uses the top 1000 significant context features for word
- Evaluate DT again

Combined Results for English

Parser	frequent	rare
Stanford (Supervised)	0.280	0.209
Seginer	0.240	0.155
Søgaard	0.227	0.144
Seginer & Søgaard	0.248	0.162
Stanford & Bigram & Trigram	†0.290	†0.217
Stanford & Seginer & Søgaard	†0.291	†0.217
Stanford & Seginer & Søgaard & Bigram & Trigram	†0.290	†0.218

- Combining UPs improves the quality of an DT
- Combining UPs with supervised parser improves the quality even more

Combined Results for German

Parser	frequent	rare
Mate (Supervised)	0.204	0.090
Seginer	0.208	0.091
Søgaard	0.193	0.077
Seginer & Søgaard	†0.218	†0.097
Mate & Bigram & Trigram	0.204	0.091
Mate & Seginer & Søgaard	†0.222	†0.10
Mate & Seginer & Søgaard & Bigram & Trigram	†0.222	†0.10

Conclusion

- Extrinsic evaluation method for UP
 - Ranking of UP is different than the Treebank Ranking
- Best Practice for building DTs
 - Building DTs using several features improves the quality
- UP can beat a supervised parser

Future Work

- Apply approach for different part-of-speech
- Analyze the impact of the sentence size
- What are the context features from UP not covered by supervised parser?
- Replace POS tags by unsupervised ones

Thanks for your attention

Germany's next top
parser
might be unsupervised