CogALex Workshop 2016
Invited Talk
December 12, 2016
Osaka, Japan

Vectors or Graphs?
On Differences of Representations for Distributional Semantic Models

Chris Biemann
biemann@informatik.uni-hamburg.de

Why Language is difficult ..

polysemous

synonymous

Concept Layer

Lexical Layer

He sat on the river bank and counted his dough.

She went to the bank and took out some money.

Tutorial at NAACL-HLT 2010, Los Angeles, CA, USA

Distributional Semantic Models

Stefan Evert, University of Osnabrück

1. DESCRIPTION

Distributional semantic models (DSM) -- also known as "word space" or "distributional similarity" models -- are based on the assumption that the meaning of a word can (at least to a certain extent) be inferred from its usage, i.e. its distribution in text. Therefore, these models dynamically build semantic representations -- in the form of high-dimensional vector spaces -- through a statistical analysis of the contexts in which words occur. DSMs are a promising technique for solving the lexical acquisition bottleneck by unsupervised learning, and their distributed representation provides a cognitively plausible, robust and flexible architecture for the organisation and processing of semantic information.

Course at ESSLLI 2016

Distributional Semantics – A Practical Introduction

Stefan Evert

- Area: LaCo
- Level: I
- Week: 1
- Time: 14:00 – 15:30
- Room: D1.02

> News: slides/handout for day 2 now available with additional code examples

Abstract

Distributional semantic models (DSM) – also known as "word space" or "distributional similarity" models – are based on the assumption that the meaning of a word can (at least to a certain extent) be inferred from its usage, i.e. its distribution in text. Therefore, these models dynamically build semantic representations of words or other linguistic units in the form of high-dimensional vector spaces, based on a statistical analysis of their distribution across documents, their collocational profiles, their syntactic dependency relations, and other contextual features. DSMs are a promising technique for solving the lexical acquisition bottleneck by unsupervised learning, and their distributed representation provides a cognitively plausible, robust and flexible architecture for the organisation and processing of semantic information.

Intro Class on “Distributional Semantics” at UT Austin
by Marco Baroni and Gemma Boleda

https://www.cs.utexas.edu/~mooney/cs388/slides/dist-sem-intro-NLP-class-UT.pdf

Distributional semantic models (DSMs)

Narrowing the field

- Idea of using corpus-based statistics to extract information about semantic properties of words and other linguistic units is extremely common in computational linguistics
- Here, we focus on models that:
  - Represent the meaning of words as *vectors* keeping track of the words' distributional history
  - Focus on the notion of *semantic similarity*, measured with geometrical methods in the *space* inhabited by the distributional vectors
  - Are intended as *general-purpose* semantic models that are estimated once, and then used for various semantic tasks, and not created ad-hoc for a specific goal
    - It follows that model estimation phase is typically unsupervised
- E.g.: LSA (Landauer & Dumais 1997), HAL (Lund & Burgess 1996), Schütze (1997), Sahlgren (2006), Padó & Lapata (2007), Baroni and Lenci (2010)
- Aka: vector/word space models, semantic spaces

Core Idea of Distributional Semantic Models:

- Collect global contexts for all words in a corpus
- Make a distributional model out of it

context

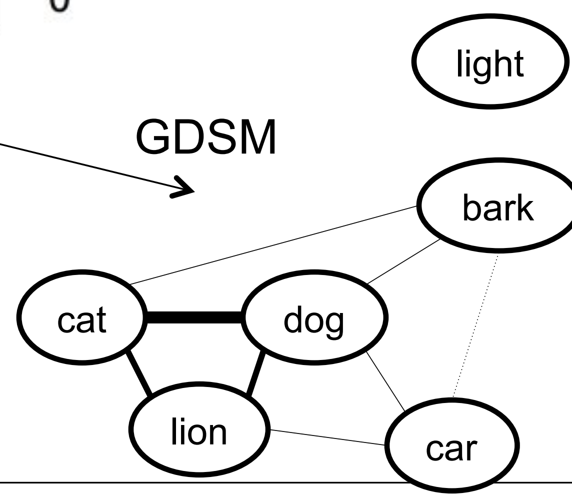
word	leash	walk	run	owner	pet	bark
dog	3	5	2	5	3	2
cat	0	3	3	2	3	0
lion	0	3	2	0	1	0
light	0	0	0	0	0	0
bark	1	0	0	2	1	0
car	0	0	1	3	0	0

sparse VDSM

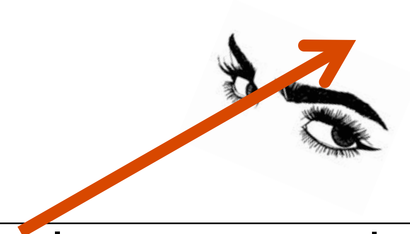
dense VDSM

	d1	d2	d3
dog	0.22	0.75	-0.31
cat	0.24	0.52	-0.05
lion	0.27	0.55	-0.12
light	-0.82	-0.13	0.02
bark	0.10	-0.04	-0.43
car	0.35	0.29	0.86

GDSM



What makes vectors so attractive?



- **The metaphor!** vector spaces allow to define distances, closeness, and can be imagined easily
- **The tradition!** Information Retrieval uses VSMs for over 40 years!
- **The mathematics!** It is straightforward to compress VSMs into dense vector spaces using PCA, SVD, etc.

Why dense vectors? (LSA, LDA, w2v, ...)

- A solution to Plato’s problem (Derweester et al., 1990) – rather not.
- A convenience for toolkits – rather yes.
- Size of the representation? – depends.

Advances of neural methods:

- fast approximation of SVD, see (Levy and Goldberg, 2014)
- there is w2v, well-engineered, and it’s really fast!
- we can tune a lot of parameters!

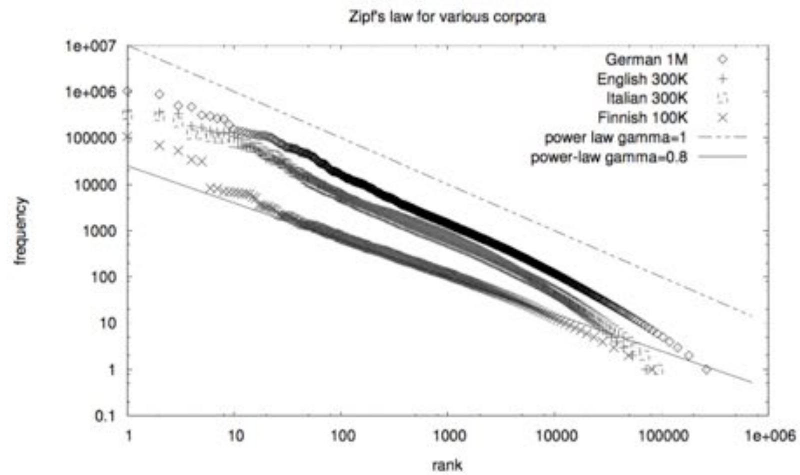
Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6):391–407

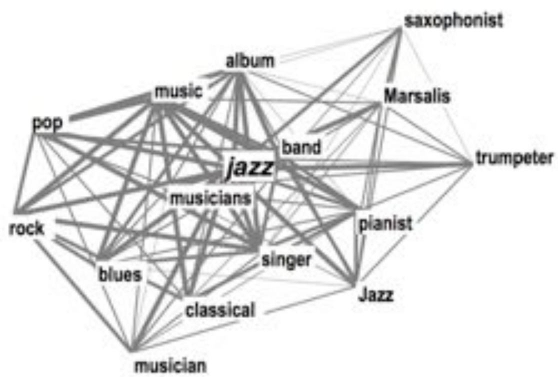
Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. Proc. NIPS 27:2177–2185

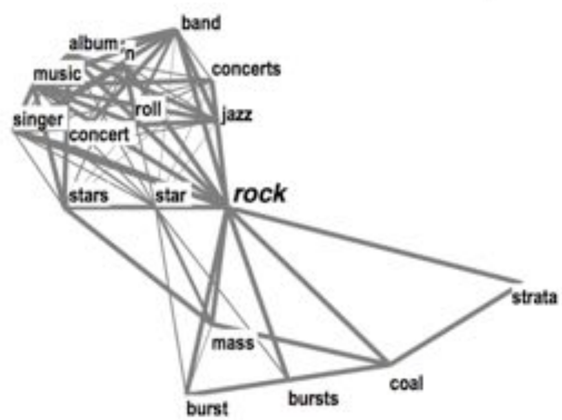
The Fallacy of Dimensionality (I)

Language is a naturally grown system:

- power-law distribution
- scale-free small-world network structure
- ‘infinite’ number of dimensions / a fractal dimension?





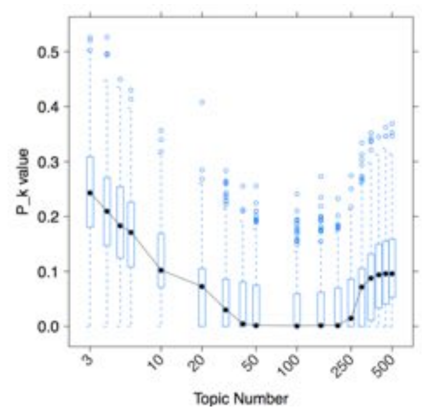


George K. Zipf. 1949. Human Behavior and the Principle of Least-Effort. Addison-Wesley, Cambridge, MA.
Mark Steyvers and Joshua B. Tenenbaum. 2005. The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth. Cognitive Science, 29(1):41–78.

The Fallacy of Dimensionality (II)

Dense Vector Spaces:

- fixed number of dimensions
- different number of optimal dimensions (from ~50 to ~ 2'000)
- necessarily lossy, like a pixel resolution: minor distinctions cannot be represented below the 'pixel size' threshold
- Two possible outcomes when optimizing the number of dimensions for a task:
  - sweet spot for number of dimensions. This is task-dependent
  - the more the better. Suggesting that no dimensionality reduction would have been even better!



Riedl, M., Biemann, C. (2012): Text Segmentation with Topic Models. Journal for Language Technology and Computational Linguistics (JLCL), 27(1):47-70

**In language, there is
no general 'right' number of dimensions!**

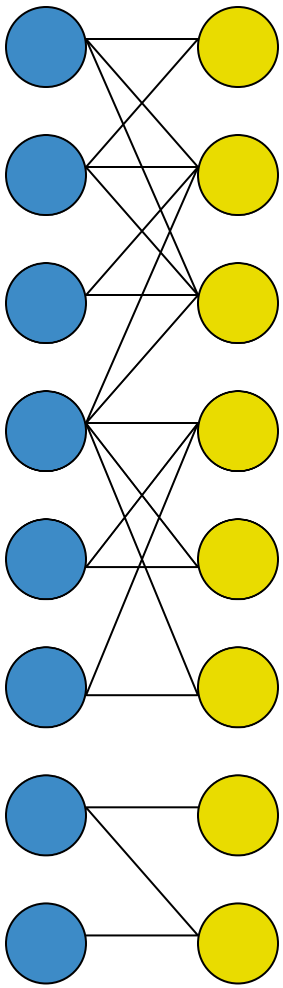
Desired Properties of Distributional Semantic Models

- Word Similarity
- Similarity and Semantic Neighborhood Computation
- Word Sense Representations
- Word Analogy and other Arithmetic
- Semantic Compositionality
- Interpretability and Robustness of Representation
- Learnability and Cognitive Plausibility

The G(V,E) View

Sources:

- words in sequence
- words in grammatical relations
- queries and clicks
- hyperlinks / citation
- ...



Parameters:

- edge weight
- node weight
- frequency threshold
- ...

JoBimText: A scalable framework for graph-based distributional semantics

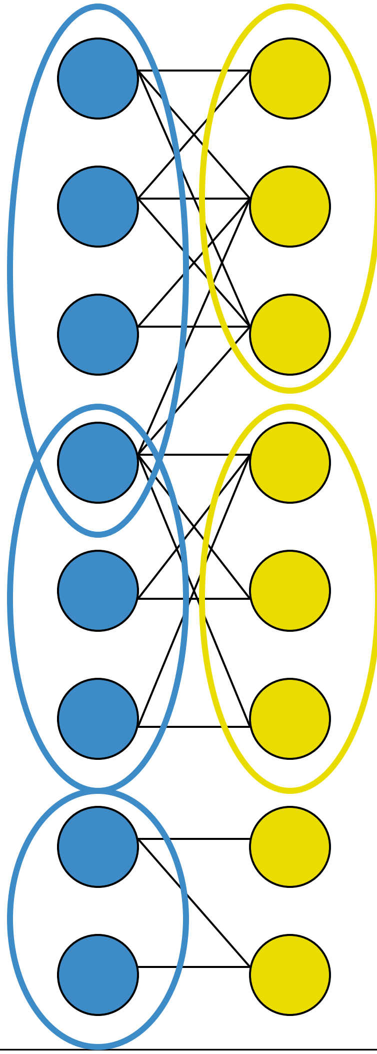
www.jobimtext.org

- Distributional semantic model: represents lexical items by their corpus-wide contexts
  - sparse representation: only retain the most significant N (e.g. 1000) contexts ('Bims') for item ('Jo')
    - **fixed length representation!**
    - cut-off reduces noise
  - context defined by 'holing system'
- scalable implementation on Apache Hadoop / Apache Spark: e.g. compute word similarities on Google Books syntactic n-grams well under a day
- open source

Biemann, C. and Riedl, M. (2013): Text: Now in 2D! A Framework for Lexical Expansion with Contextual Similarity. Journal of Language Modeling 1(1):55-95

Similarity

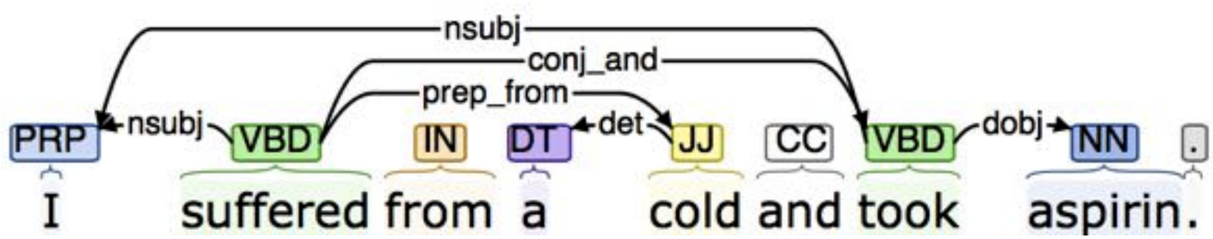
- Similarity as function of shared contexts / common features



- Graph clustering makes similarity of item sets explicit

The @ 'holing' operation: producing pairs of words and contexts

SENTENCE:



STANFORD COLLAPSED DEPENDENCIES: *http://nlp.stanford.edu:8080/parser/*

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

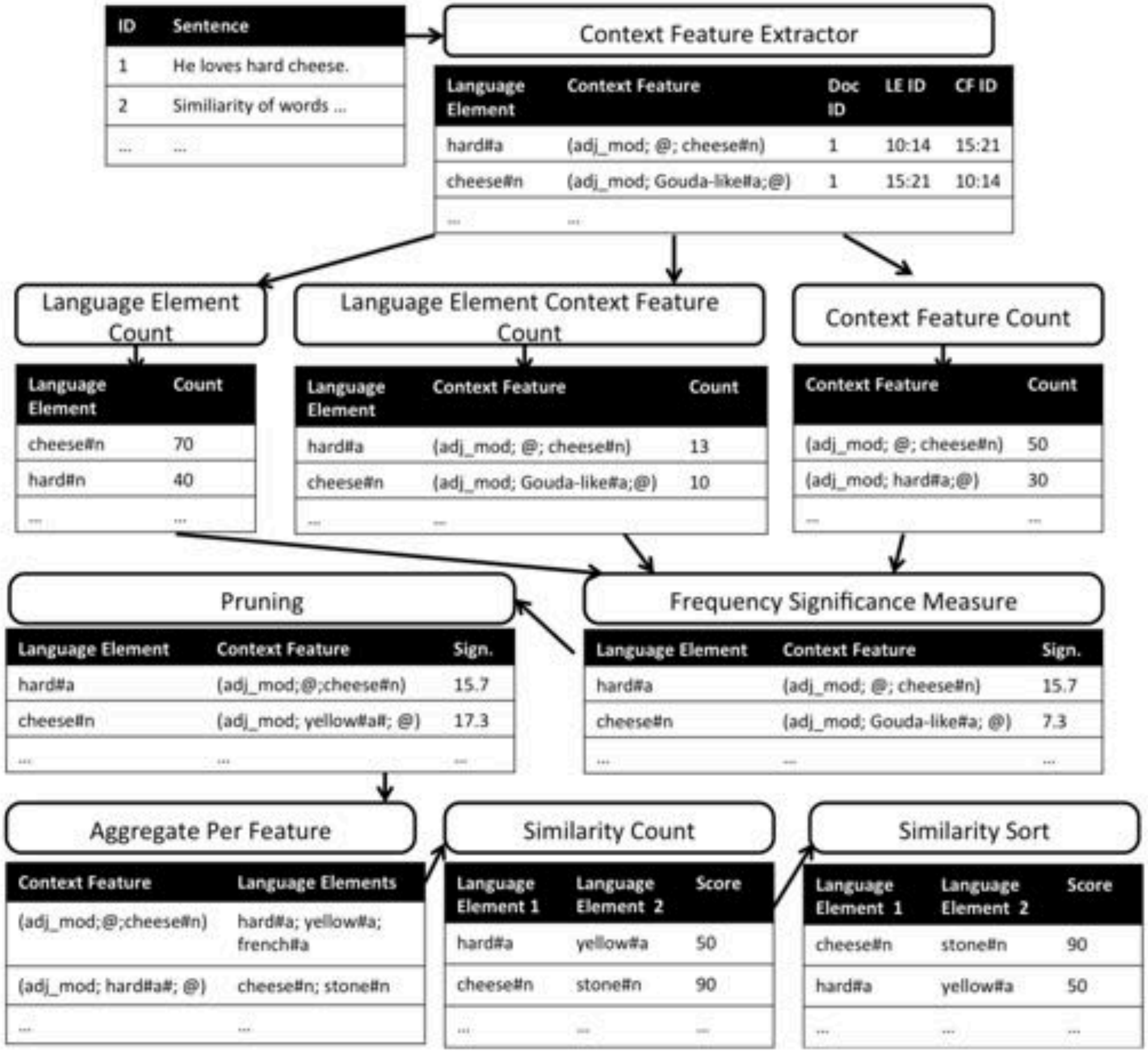
WORD-CONTEXT PAIRS:

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

Scaling Computation with MapReduce

- read: this scales somehow without using a lot of RAM

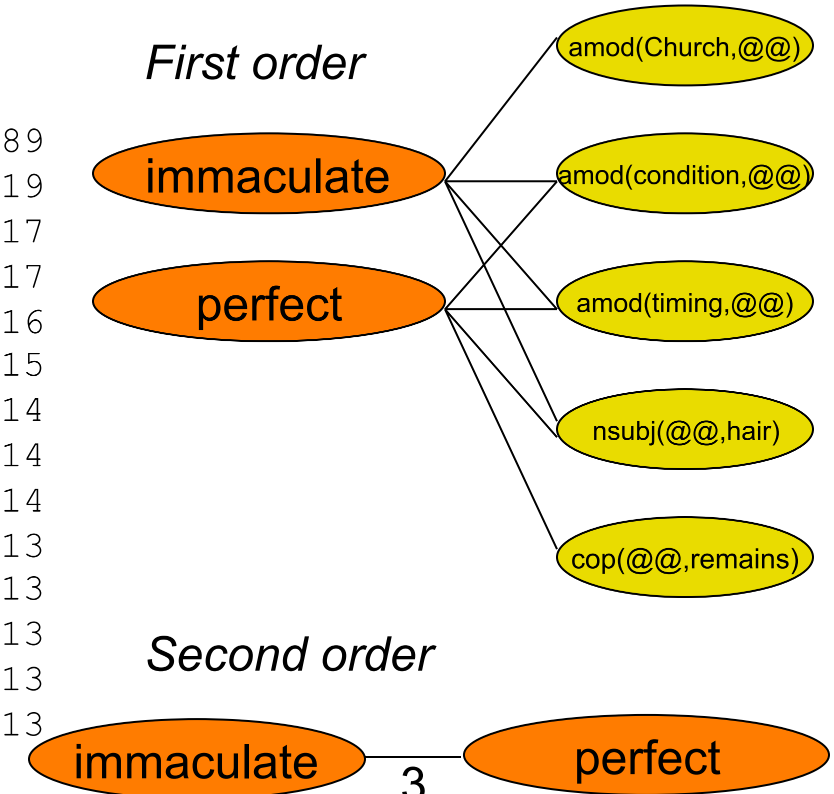


Distributional Thesaurus (DT)

- Computed from distributional similarity statistics
- **Entry** for a **target** word consists of a ranked list of neighbors

meeting	
meeting	288
meetings	102
hearing	89
session	68
conference	62
summit	51
forum	46
workshop	46
hearings	46
ceremony	45
sessions	41
briefing	40
event	40
convention	38
gathering	36
...	

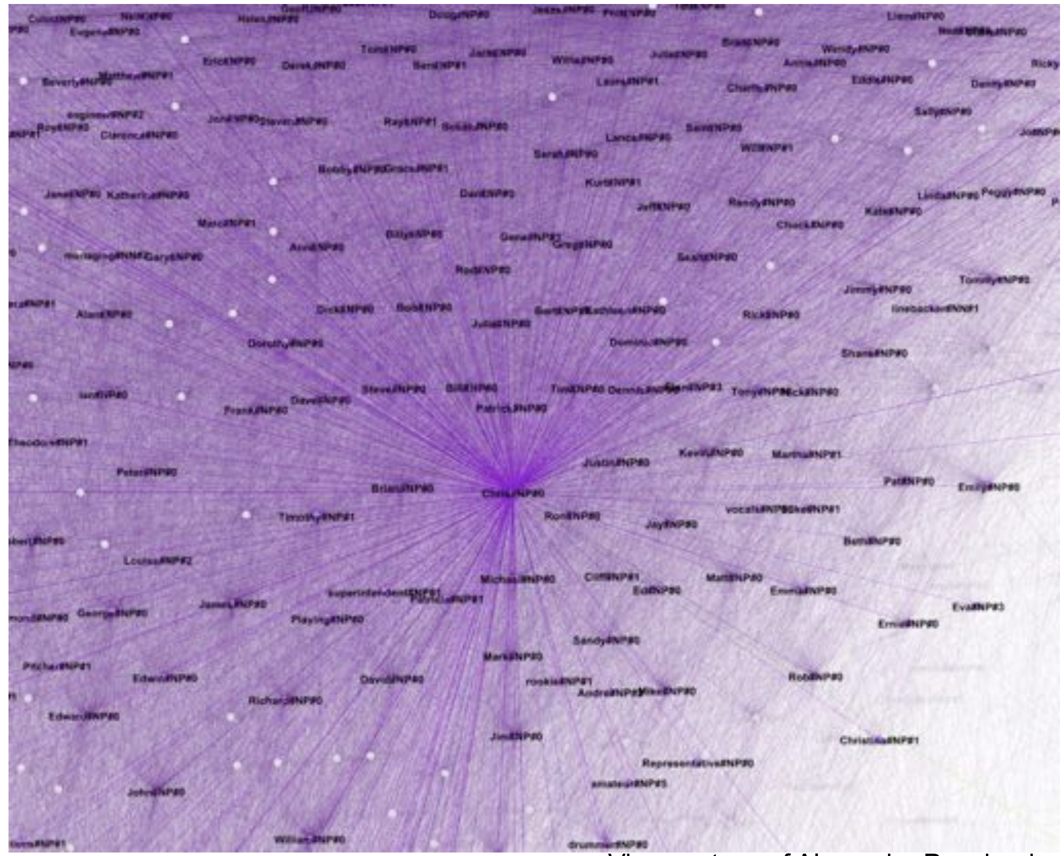
articulate	
articulate	89
explain	19
understand	17
communicate	17
defend	16
establish	15
deliver	14
evaluate	14
adjust	14
manage	13
speak	13
change	13
answer	13
maintain	13
...	



Dekang Lin. 1998. Automatic Retrieval and Clustering of Similar Words. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2, pages 768–774, Montreal, QC, Canada.

Graph Structure of Lin's Distributional Thesaurus

```
duty
|___responsibility 0.21 0.21
|    |___role 0.12 0.11
|    |    |___action 0.11 0.10
|    |         |___change 0.24 0.08
|    |         |    |___rule 0.16 0.08
|    |         |         |___restriction 0.27 0.08
|    |         |         |    |___ban 0.30 0.08
|    |         |         |    |___sanction 0.19 0.08
|    |         |         |___schedule 0.11 0.07
|    |         |         |___regulation 0.37 0.07
|    |         |___challenge 0.13 0.07
|    |         |    |___issue 0.13 0.07
|    |         |         |___reason 0.14 0.07
|    |         |         |___matter 0.28 0.07
|    |         |___measure 0.22 0.07
|    |___obligation 0.12 0.10
|    |___power 0.17 0.08
|    |    |___jurisdiction 0.13 0.08
|    |    |___right 0.12 0.07
|    |    |___control 0.20 0.07
|    |    |___ground 0.08 0.07
|    |___accountability 0.14 0.08
|    |___experience 0.12 0.07
|___post 0.14 0.14
|    |___job 0.17 0.10
|    |    |___work 0.17 0.10
|    |         |___training 0.11 0.07
|    |___position 0.25 0.10
|___task 0.10 0.10
|    |___chore 0.11 0.07
|___operation 0.10 0.10
|    |___function 0.10 0.08
|    |___mission 0.12 0.07
|    |    |___patrol 0.07 0.07
|    |___staff 0.10 0.07
|___penalty 0.09 0.09
|    |___fee 0.17 0.08
|         |___tariff 0.13 0.08
|         |___tax 0.19 0.07
|___reservist 0.07 0.07
```



Viz. courtesy of Alexander Panchenko

Dekang Lin. 1998. Automatic Retrieval and Clustering of Similar Words. In Proceedings of COLING/ ACL 1998, pages 768–774, Montreal, QC, Canada.

Word Similarity

Graph-based DSM:

- explicitly stores top-n similar words in a graph
- explicitly stores features, easy to retrieve common features
- words that share few or no fatures cannot be compared

Vector-based DSMs:

- words are points in a vector space.
- If dense: dimensions do not mean anything, information on common features is lost
- any pair of words can be compared

What is more related: **rooster:voyage** or **asylum:fruit** ?

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. Communications of the ACM, 8(10):627–633.

Word Similarity

Graph-based DSM:

- explicitly stores top-n similar words in a graph
- explicitly stores features, easy to retrieve common features
- words that share few or no fatures cannot be compared

Vector-based DSMs:

- words are points in a vector space.
- If dense: dimensions do not mean anything, information on common features is lost
- any pair of words can be compared

What is more related: **rooster:voyage** or **asylum:fruit** ?
0.04 0.19

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. Communications of the ACM, 8(10):627–633.

Semantic Neighborhoods

Graph-based DSM:

- directly retrieve most similar items from similarity graph
- limited amount of similar items, either by top-n or by threshold on common features
- asymmetric mutual ranks: no such thing as the triangle inequality

Vector-based DSM:

- neigborhood search is expensive, needs engineering like K-D-trees
- pre-computation of top-n similar is possible but does not scale well
- triangle inequality holds: distance(a,c) ≤ distance (a,b) + distance (b,c).

Python

python	324
snake	112
serpent	91
rattlesnake	72
cobra	72
dragon	68
crocodile	63
alligator	59
tiger	55
viper	53
constrictor	52
lion	48
leopard	48
shark	42
lizard	41
panther	41
adder	41
elephant	40
reptile	40
jaguar	39
bear	37
wolf	37
tortoise	36
monster	36
anaconda	36

Anaconda

anaconda	107
python	36
snake	31
serpent	26
cobra	25
constrictor	24
boa	23
rattlesnake	23
viper	21
crocodile	19
alligator	19
adder	18
dragon	17
tiger	14
snake	14
monster	13
reptile	13
wolf	11
worm	9
leopard	9
whip	9
vulture	9
toad	8
rattler	8
panther	8

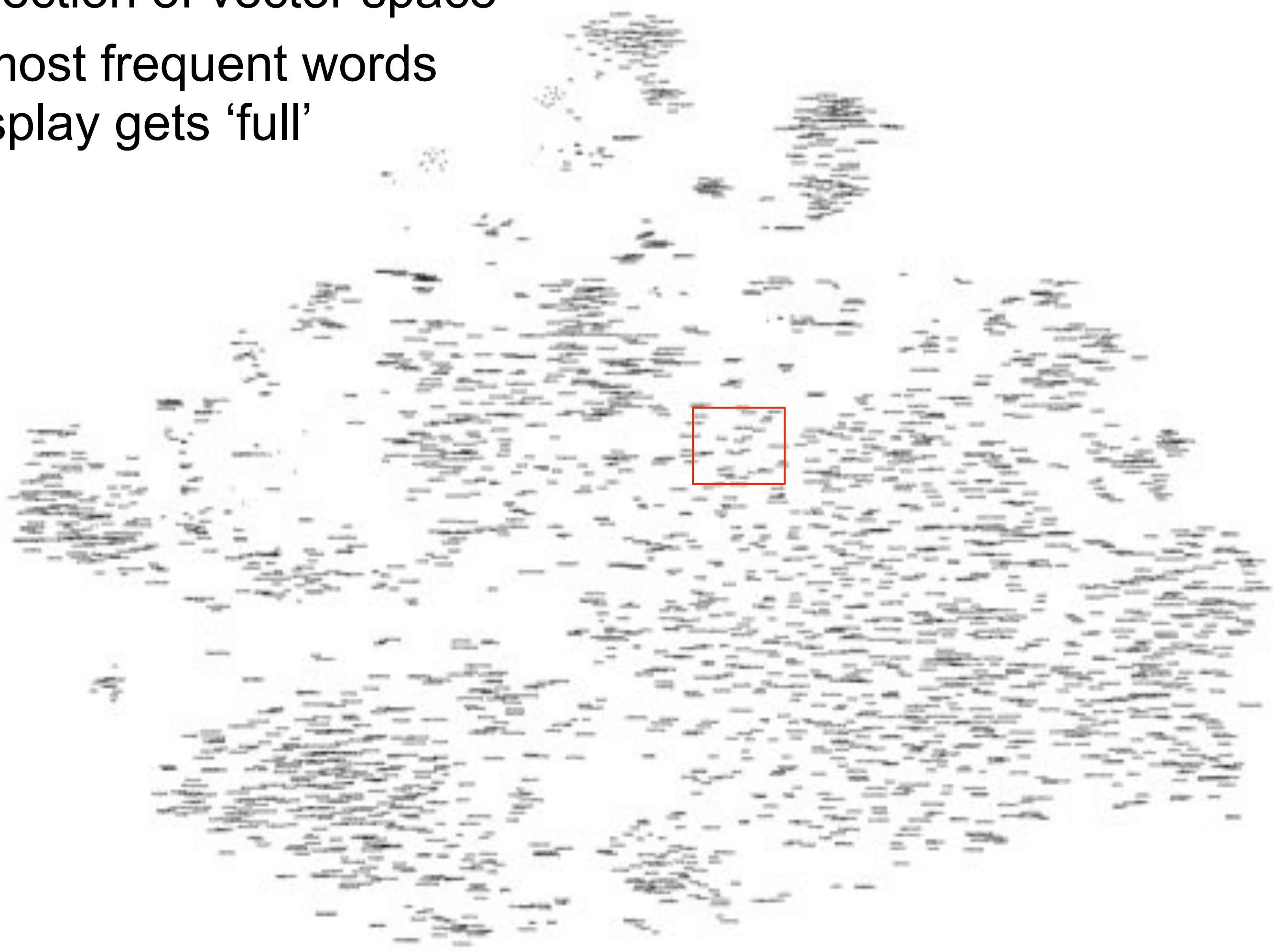
www.jobimtext.org/jobimviz

Kohei Sugawara, Hayato Kobayashi, and Masajiro Iwasaki. 2016. On approximately searching for similar word embeddings. Proc. ACL 2016, pages 2265–2275, Berlin, Germany

Zoom in ...

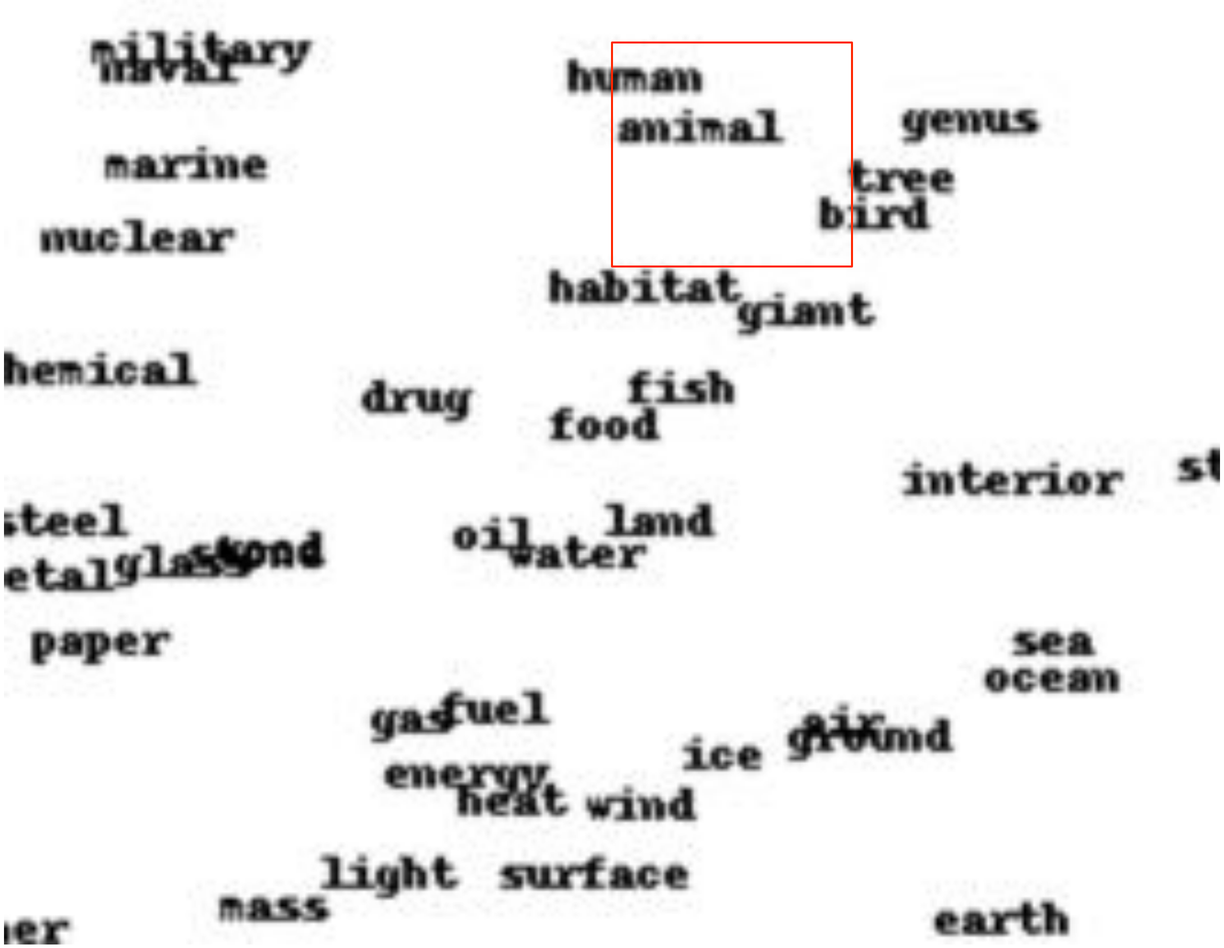
http://www.cs.toronto.edu/~hinton/turian.png

- 2D-projection of vector space
- Show most frequent words until display gets ‘full’

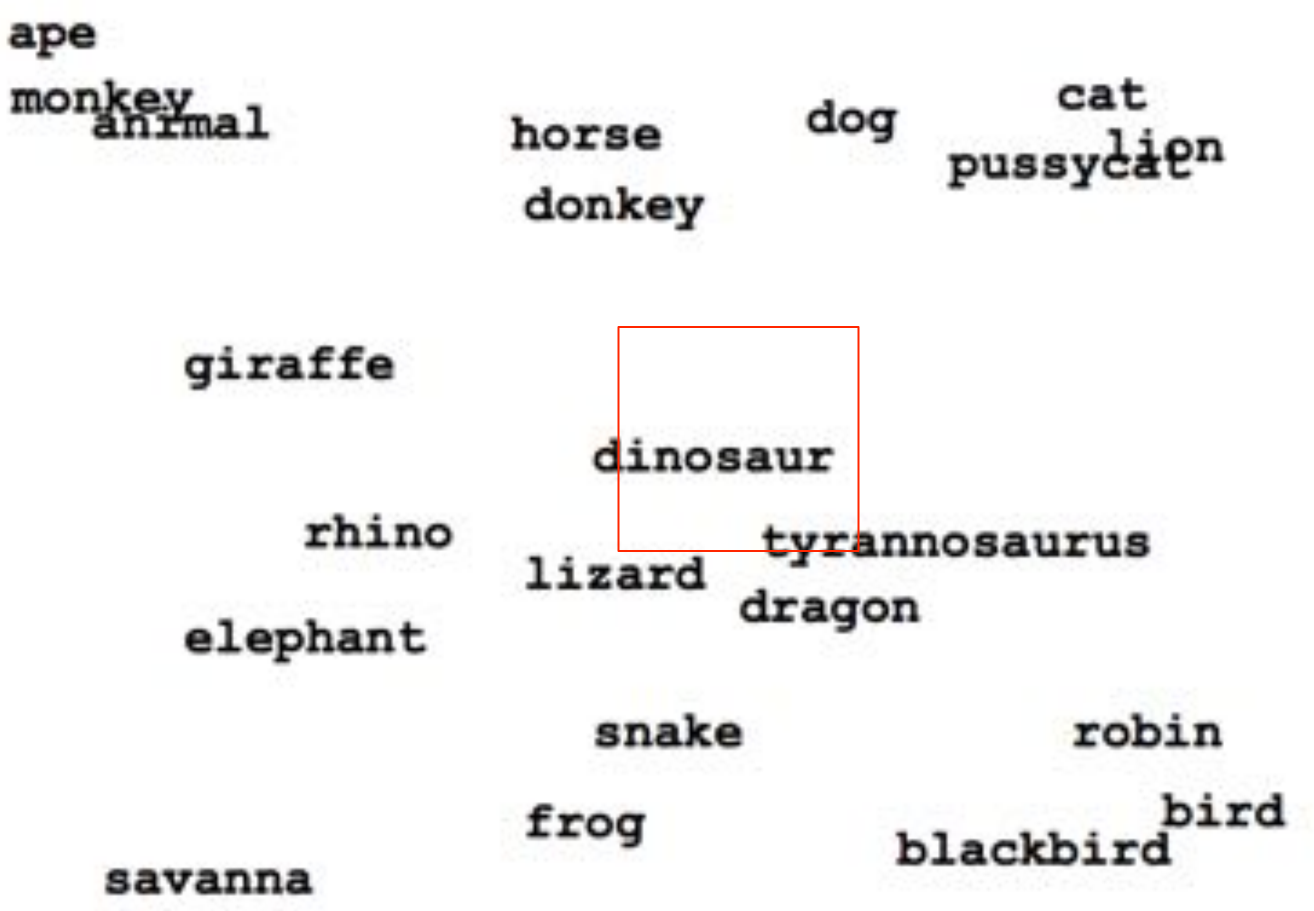


Zoom in ...

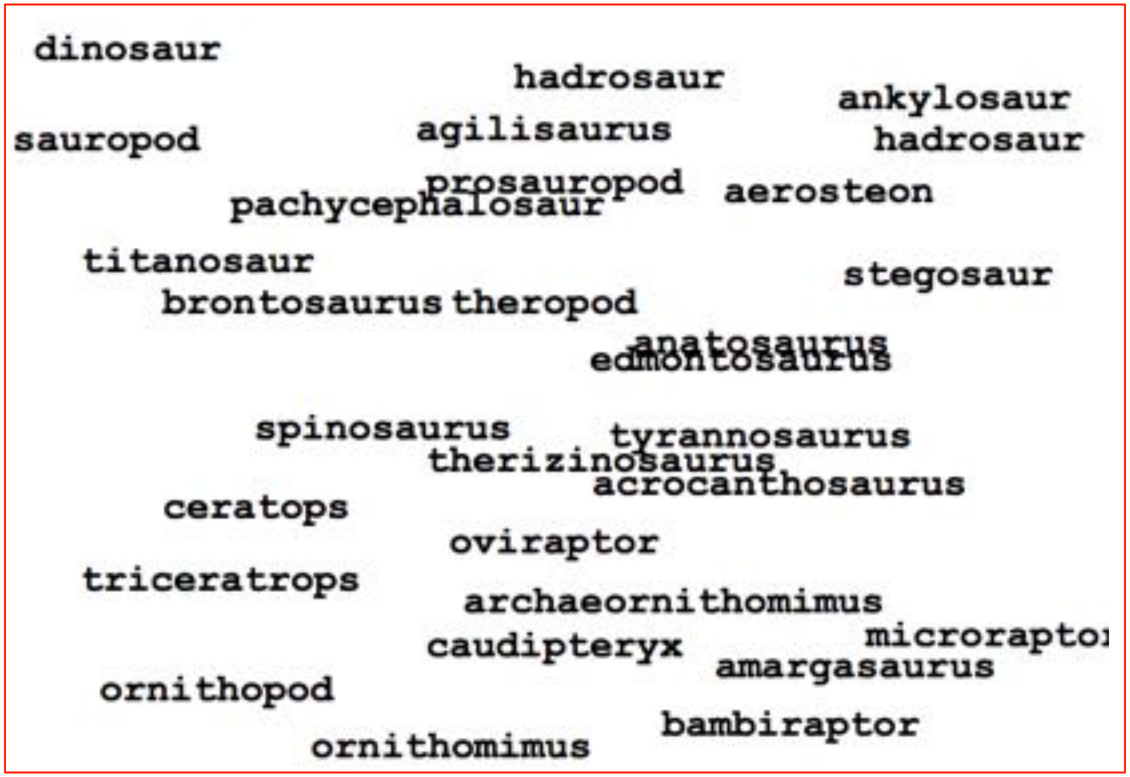
http://www.cs.toronto.edu/~hinton/turian.png



Zoom in ...


ape
monkey
animal
horse
dog
cat
lion
pussycat
donkey
giraffe
dinosaur
rhino
tyrannosaurus
lizard
dragon
elephant
snake
robin
frog
bird
blackbird
savanna

Now, return the semantic neighborhood!



- Most neighbors are rare: no notion of frequency in VDSM
- How large must neighborhood grow to discover 'prototypes'? e.g.
  - bambiraptor ISA
  - dinosaur ISA
  - animal

Desirable? Depends on the task!

Sample Application: OOV replacement

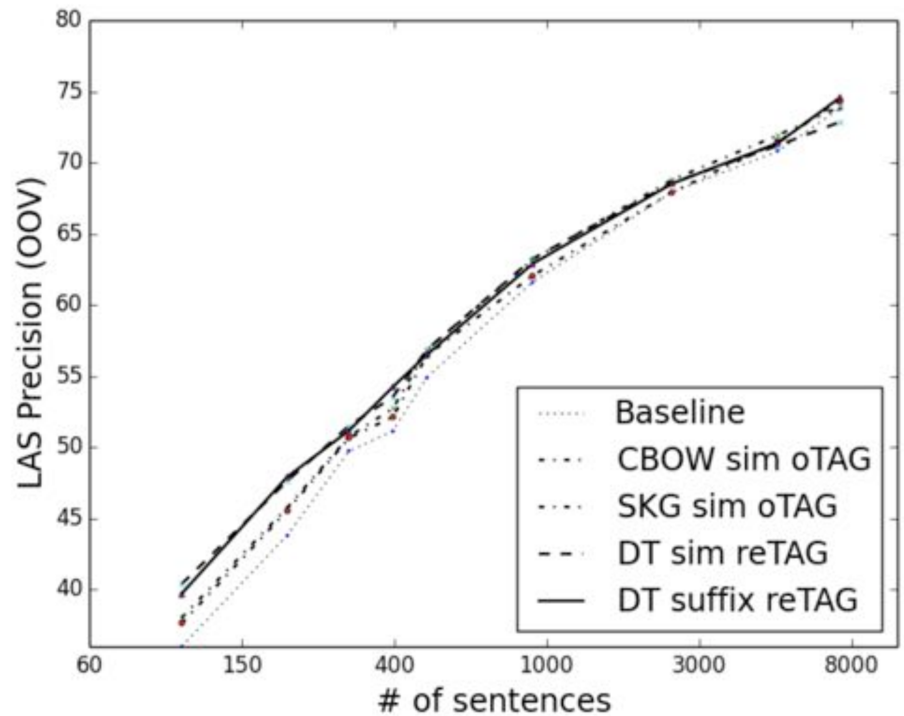
- Say you have a tagger or parser that has a hard time with out-of-vocabulary words (ALL supervised taggers/parsers)
- Say you do not want to re-train it – can you still improve it?
- OOV replacement: replace OOV words with most similar word from a DSM that is in-vocabulary
  - baseline: use first word with longest suffix overlap from training
  - sim: use most similar in-vocabulary word
  - suffix: of the words with longest suffix overlap, choose the most similar one

LANG	OOV	baseline		suffix only		DT sim		DT suffix	
	%	all	OOV	all	OOV	all	OOV	all	OOV
Arabic	10.3	**98.53**	**94.01**	97.82#	87.44#	98.49#	93.67#	98.52	93.91
English	8.0	93.43	75.39	93.09#	72.03#	**93.82***	**78.67***	93.61*	76.75
French	5.3	95.47	83.29	95.17#	78.30#	95.68*	86.28*	**95.73***	**86.78***
German	11.5	**91.92**	85.63	90.88#	77.70#	91.84	85.32	**91.92**	**85.68**
Hindi	4.4	95.35	76.41	95.07#	71.27#	95.41	77.5	[illegible]	[illegible]
Spanish	6.9	94.82	79.62	95.00	81.17	95.45*	**86.3**	[illegible]	[illegible]
Swedish	14.3	95.34	89.80	94.78#	86.04 #	95.57*	90.8	[illegible]	[illegible]

LANG	SKG				CBOW			
	sim		suffix		sim		suffix	
	all	OOV	all	OOV	all	OOV	all	OOV
Arabic	98.46#	93.39#	98.50#	93.73#	98.48#	93.60#	98.52	93.94
English	93.10#	72.29#	93.57	76.31	93.24#	73.91	93.52	75.70
German	90.99#	77.65#	91.62#	83.61#	91.78	83.92#	91.91	85.43

When to say “no”? The case for OOV replacement

- advantage of *DT:* can NOT return a replacement when it has too low confidence.
- any threshold on hyper-sphere radius or number of neighbors in w2v VDSM did not change anything
- No notion of frequency: neighborhood in VDSM consists of many rare words



Prasanth Kolachina, Martin Riedl and Chris Biemann (will appear someday): Replacing OOV Words with Distributional Semantics for Dependency Parsing (submission pending)

2D Text:
Matching Meaning beyond Keywords

Where was the first professor for electric science established?

almost
no word
overlap

In 1883 the first faculty for electrical engineering was founded there.

2D Text: Matching Meaning beyond Keywords

Where was the first professor for electric science established?

professor	electric	science	established?
director	electrical	biology	create
emeritus	heavy-duty	economics	form
dean	antique	sciences	set
lecturer	battery-powered	mathematics	maintain
president	electronic	physics	found
psychologist	stainless	math	abolish
historian	diesel	psychology	strengthen

In 1883 the first faculty for electrical engineering was founded there.

faculty	electrical	engineering	founded
teacher	electric	science	co-found
professor	mechanical	sciences	form
student	thermal	biology	establish
graduate	electronic	physics	own
alumnus	industrial	economics	join
staff	optical	mathematics	rename
campus	automotive	psychology	bear

2D Text: Matching Meaning beyond Keywords

Where was the first professor for electric science established?

professor	electric	science	established
director	electrical	biology	create
emeritus	heavy-duty	economics	form
dean	antique	sciences	set
lecturer	battery-powered	mathematics	maintain
president	electronic	physics	found
psychologist	stainless	math	abolish
historian	diesel	psychology	strengthen

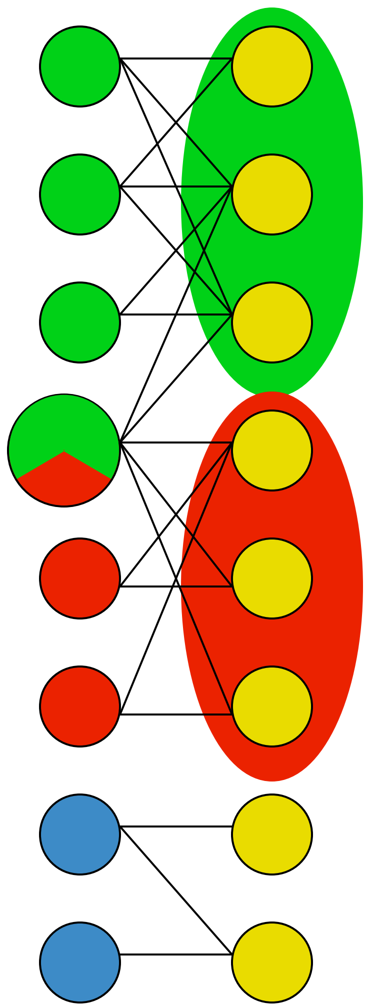
In 1883 the first faculty for electrical engineering was founded there.

faculty	electrical	engineering	founded
teacher	electric	science	co-found
professor	mechanical	sciences	form
student	thermal	biology	establish
graduate	electronic	physics	own
alumnus	industrial	economics	join
staff	optical	mathematics	rename
campus	automotive	psychology	bear

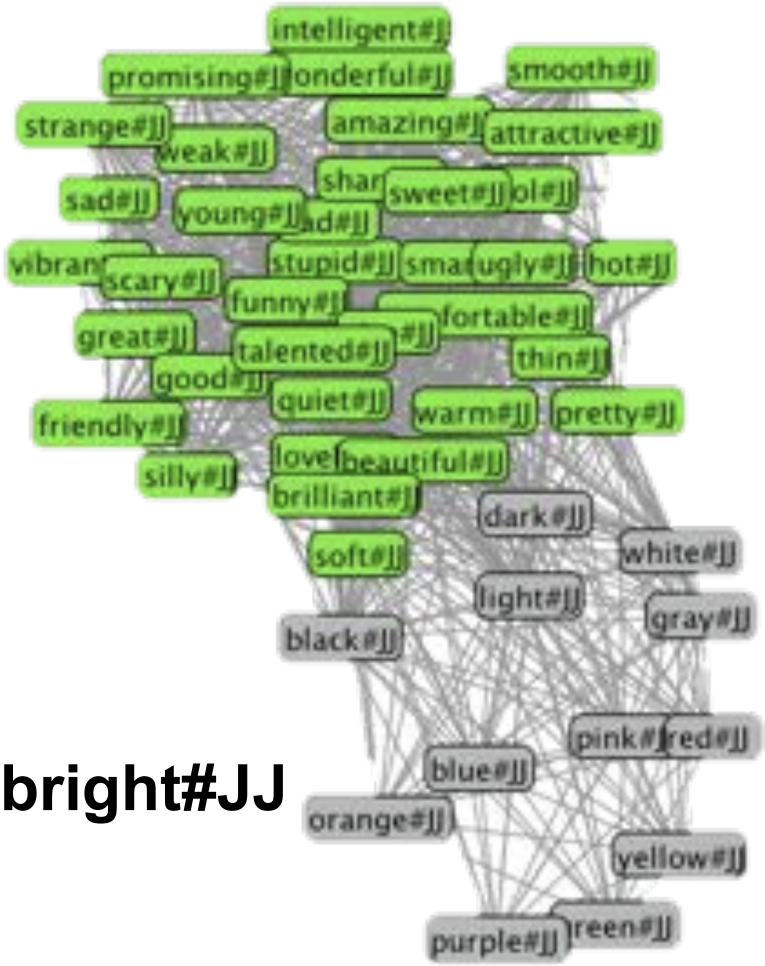
Biemann, C., Riedl, M. (2013): Text: Now in 2D! A Framework for Lexical Expansion with Contextual Similarity. Journal of Language Modelling 1(1): 55--95

Word Sense Representation

- Ambiguous items have several senses: connect to different clusters
- Estimation of sense priors

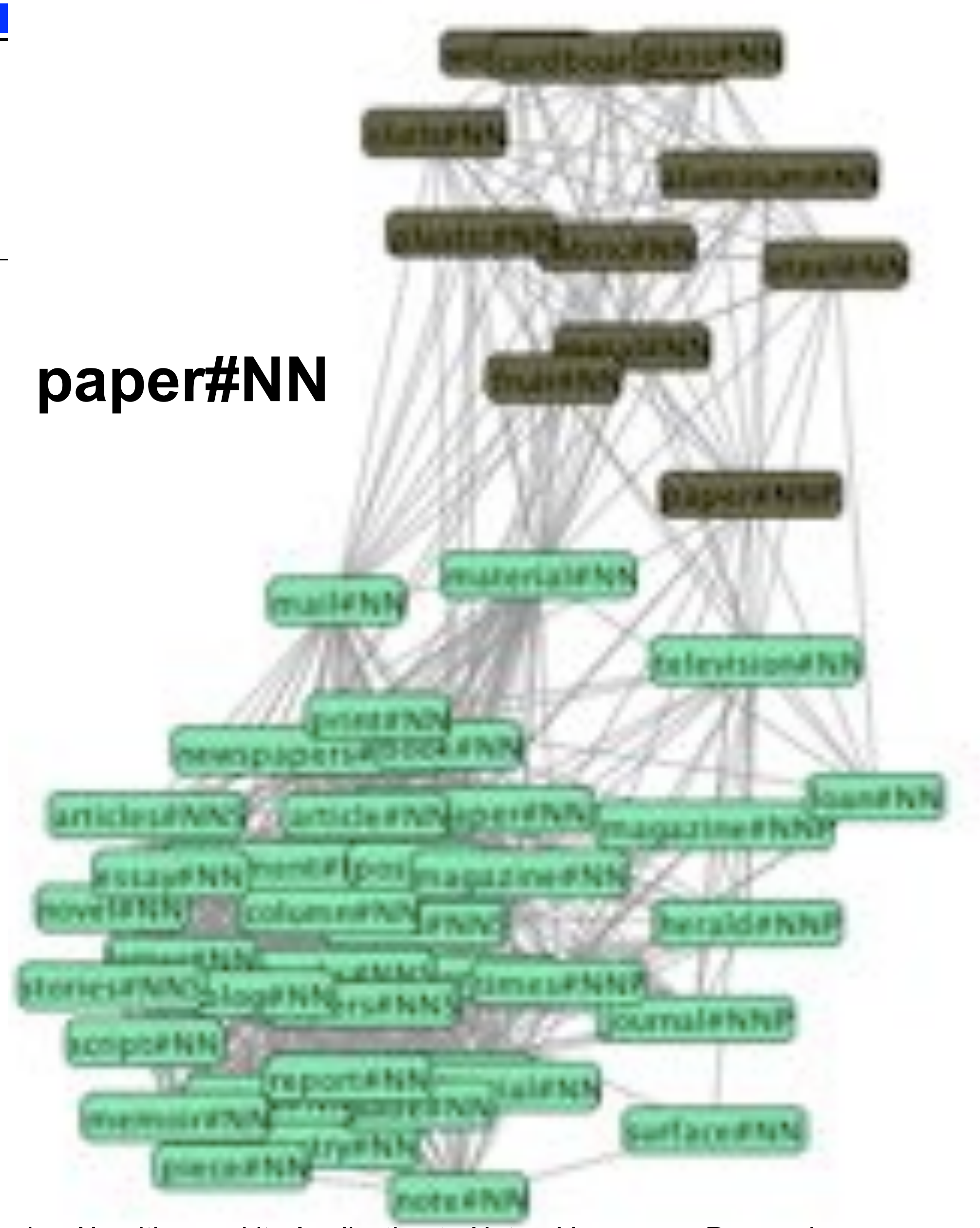


Clustering of DT entries: Sense Induction



bright#JJ

paper#NN



C. Biemann (2006): Chinese Whispers - an Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems. Proceedings of the HLT-NAACL-06 Workshop on Textgraphs-06, New York, USA.

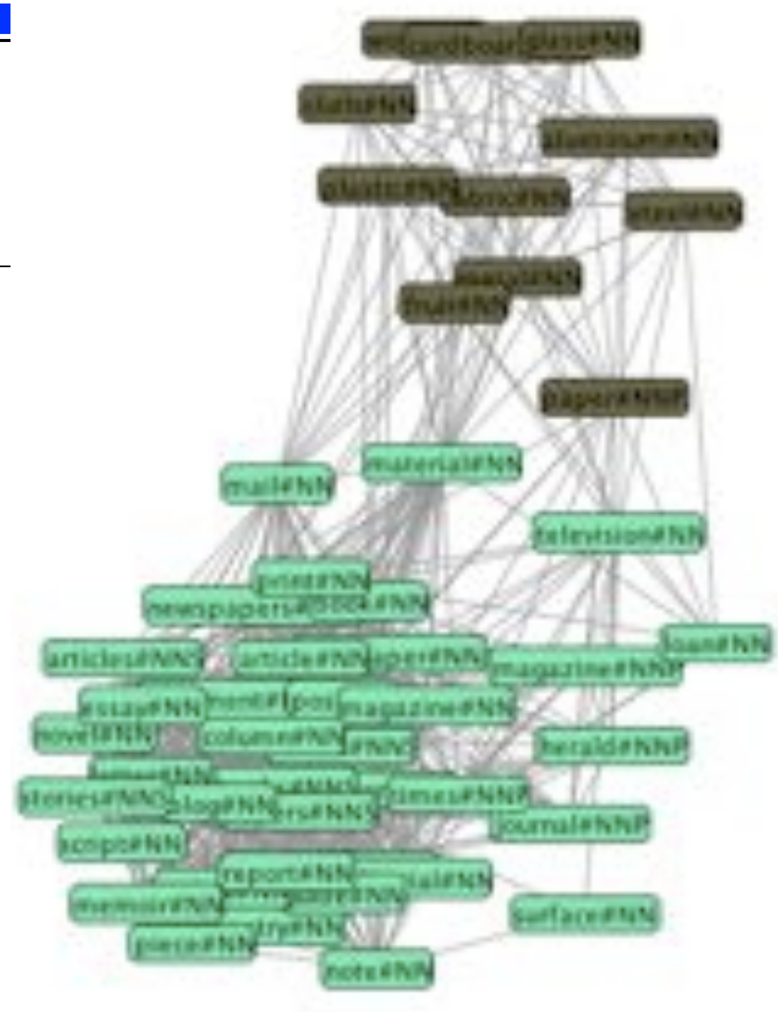
Features for Disambiguation

paper 0 (newspaper)

read#VB#-dobj	45
reading#VBG#-dobj	45
write#VB#-dobj	38
read#VBD#-dobj	37
writing#VBG#-dobj	36
wrote#VBD#-dobj	34
original#JJ#amod	27
wrote#VBD#-prep_in	26
recent#JJ#amod	26
published#VBN#partmod	25
written#VBN#-dobj	23
published#VBN#-nsubjpass	20
published#VBD#-dobj	19
copy#NN#-prep_of	18
said#VBD#-prep_in	18
author#NN#-prep_of	17
pages#NNS#-prep_of	16
told#VBD#-dobj	15
buy#VB#-dobj	14
published#VBN#-prep_in	14
page#NN#-prep_of	14

paper 1 (material)

piece#NN#-prep_of	21
pieces#NNS#-prep_of	17
made#VBN#-prep_from	13
bags#NNS#-nn	11
white#JJ#amod	9
paper#NN#-conj_and	9
glass#NN#-conj_and	9
products#NNS#-nn	9
industry#NN#-nn	8
plastic#NN#conj_and	8
plastic#NN#-conj_and	8
bits#NNS#-prep_of	8
bag#NN#-nn	8
plastic#NN#conj_or	8
sheet#NN#-prep_of	7
recycled#JJ#amod	7
tons#NNS#-prep_of	7
glass#NN#conj_and	7
buy#VB#-dobj	6
plates#NNS#-nn	6
pile#NN#-prep_of	6

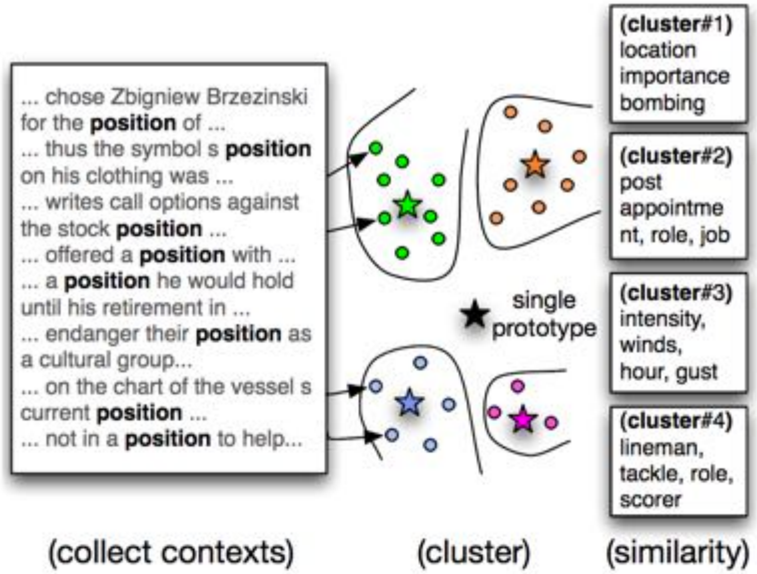


These are shared by **paper** and the cluster members.

Disambiguation: find features in context.

I am reading an original paper on the **recycled** paper **industry** .

Sense Embeddings? Yes, but ...



- Approaches relying on a knowledge base: "Use WordNet and average vectors per concept" (Rothe and Schütze, 2016, inter al).
- Unsupervised approaches with fixed K: "cluster neighborhoods with k-means" (Reisinger and Mooney, 2010, inter al.)
- Nonparametric approaches:
  - Bartunov et al., 2015
  - Neelakantan et al., 2014

Joseph Reisinger and Raymond J. Mooney. 2010. Multi-prototype vector-space models of word meaning. In Proc. NAACL-HLT 2010, Los Angeles, CA, USA, pp. 109-117.
Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In Proc. EMNLP 2014, pages 1059–1069, Doha, Qatar.
Sergey Bartunov, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. 2016. Breaking sticks and ambiguities with adaptive skip-gram. In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)
Sasha Rothe and Hinrich Schütze. 2015. AutoExtend: Extending Word Embeddings to Embeddings for Synsets
and Lexemes. Proc. ACL 2015, Beijing, China, pp. 1793-1803

Symbolic Distributional Model example “beetle”

Linking Language to Knowledge with Distributional Semantics

Sense	Hypernyms	Similar lexical items	Aggregated Context Clues
beetle.0	car, company, macho nameplate, nameplate, icon, hit	camaro, mustang, gto, corvette, convertible, oldsmobil, minivan, camry, corolla, vw, impala, gt, thunderbird, jetta, convertible, gti, passat, sedan	<nn:car <nn:model <nn:dealership <nn:brand <nsubj:sell <dobj:drive <nsubj:have <nn:dealer <nn:owner <nn:vehicle <dobj:buy <nn:sale <nn:engine <nn:executive <nsubj:play >possessive:'s <nn:driver <nn:coupe <nsubj:offer <appos:car <dobj:own <nsubj:make <nsubj:announce <conj_and:bmw <poss:model <nn:convertible <nsubj:introduce >conj_and:bmw <nn:automobile <nsubj:car <nn:plant <nn:wagon <nn:engineer (...)
beetle.1	animal, species, insect, wildlife, creature	amphibian, bug, pythons, alligator, earwig, reptile, frog, bird, crocodile, wasp, grasshopper, earthworm, (.. 114 more) ..., worm, butterfly, ladybug, parrot, gecko, cutworm, weevil, salamander, lemur	>det:the <dobj:kill <nsubj:are >det:these <dobj:find <nsubjpass:find >conj_and:insect >det:some <dobj:eat >det:a <prep_of:rid <nsubj:feed <dobj:keep <prep_of:species <dobj:call <nsubj:spread >amod:tiny <dobj:see <prep_of:type <conj_and:insect <prep_of:presence >det:those <prep_with:infested >cop:are <dobj:control <prep_of:number

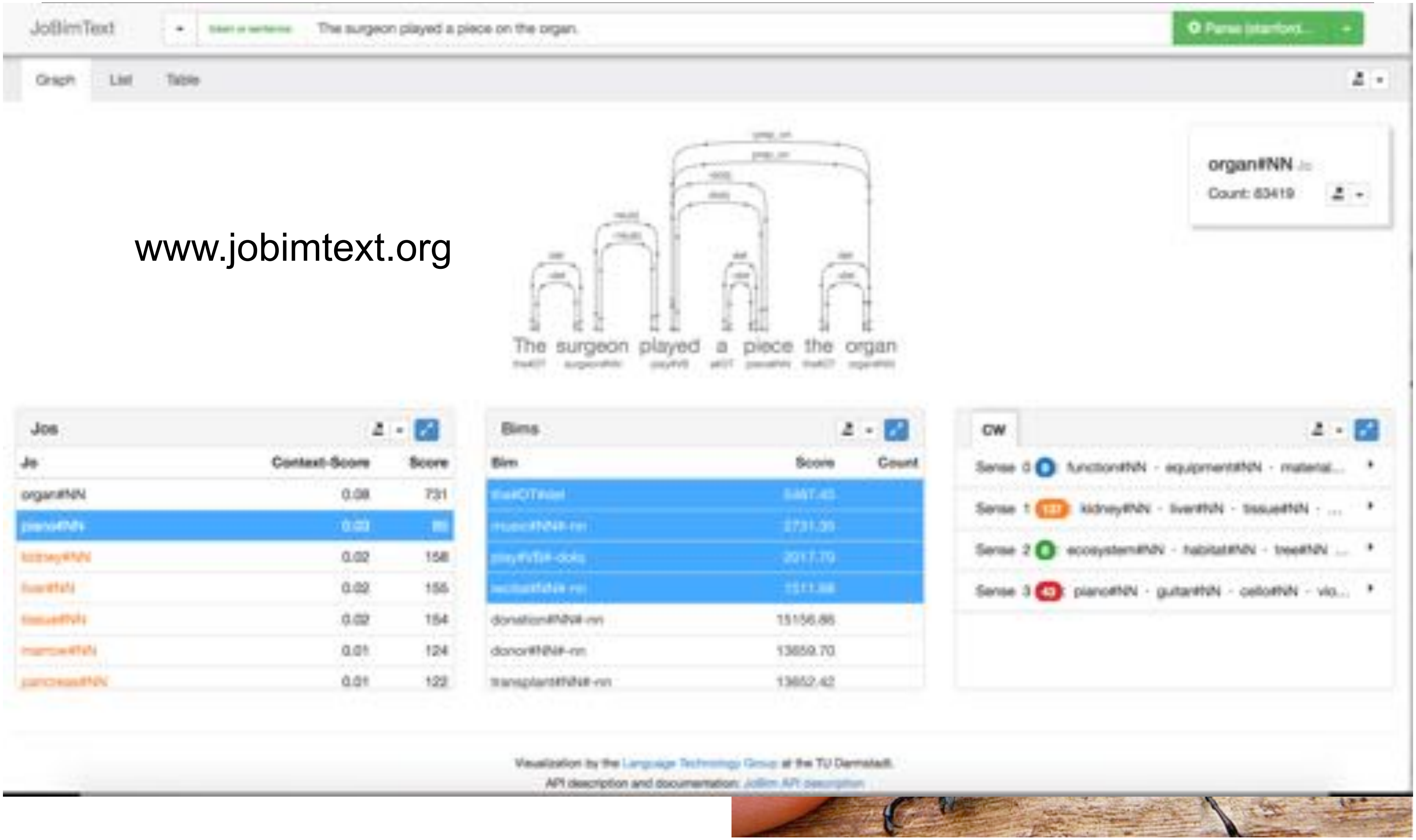
Biemann, C. and Riedl, M. (2013): Text: Now in 2D! A Framework for Lexical Expansion with Contextual Similarity. Journal of Language Modeling 1(1):55-95

Symbolic Distributional Model example “beetle”

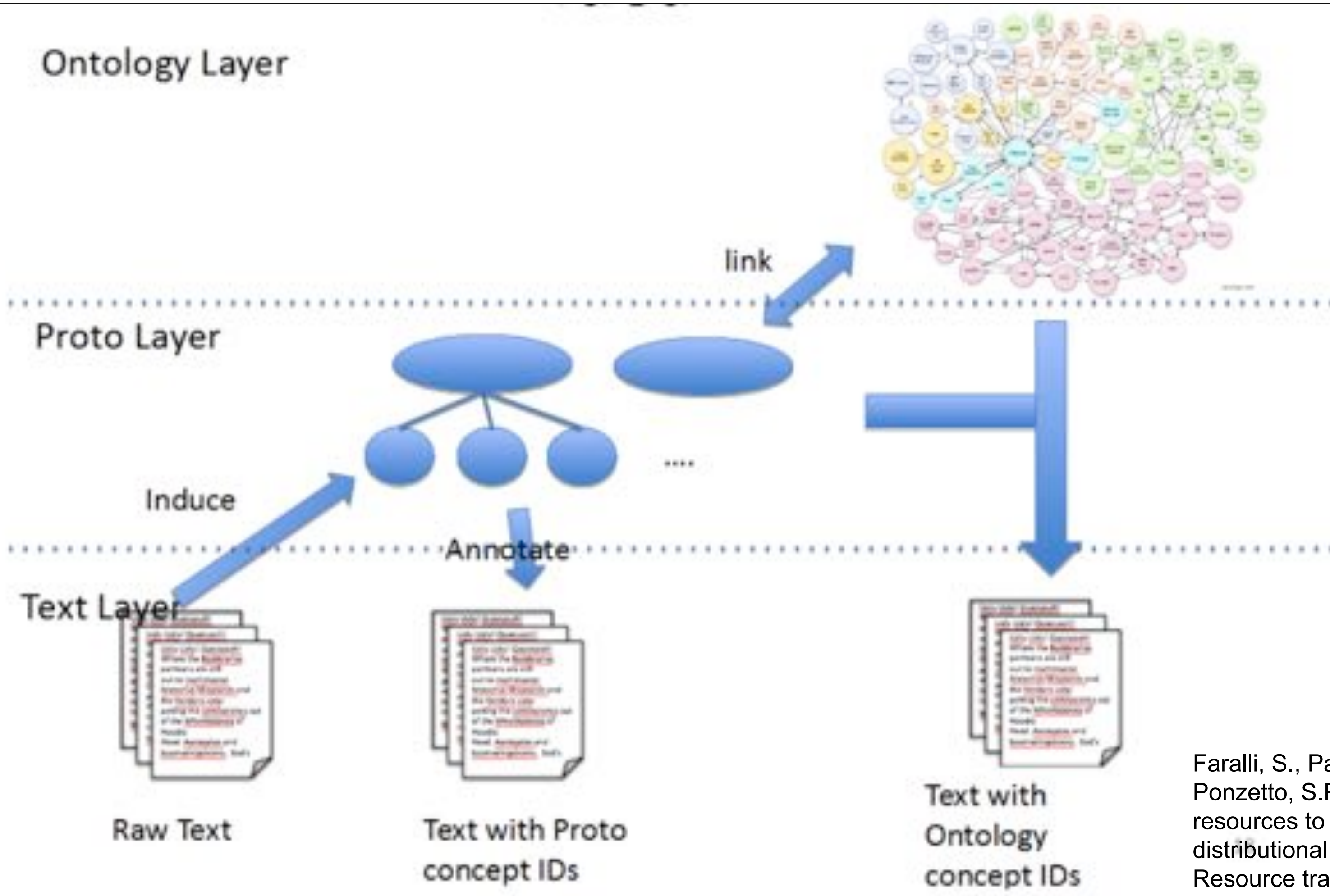
JoBimText

Linking Language to Knowledge with Distributional Semantics

www.jobimtext.org



Joining Ontologies and semantics INduced from Text (JOIN-T)



Faralli, S., Panchenko, A., Biemann, C., Ponzetto, S.P. (2016): Linking lexical resources to disambiguated distributional semantic networks. ISWC Resource track 2016, Kobe, Japan

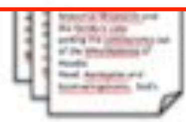
Joining Ontologies and semantics INduced from Text (JOIN-T)

Ontology Layer

mouse and keyboard JoBimText model entries			
entry	similar terms	hypernyms	context clues
mouse:NN:0	rat:NN, rodent:NN, monkey:NN, ...	animal:NN, species:NN, ...	rat::NN:conj_and, white-footed:JJ:amod, ...
mouse:NN:1	keyboard:NN, computer:NN, printer:NN ...	device:NN, equipment:NN, ...	click:NN:-prep_of, click:NN:-nn, ...
keyboard:NN:0	piano:NN, synthesizer:NN, organ:NN ...	instrument:NN, device:NN, ...	play:VB:-dobj, electric:JJ:amod, ..
keyboard:NN:1	keypad:NN, mouse:NN, screen:NN ...	device:NN, technology:NN ...	computer:NN:nn, qwerty:JJ:amod ...

mouse and keyboard PCZ proto-concepts			
entry	similar terms	hypernyms	context clues
mouse:NN:0	rat:NN:0, rodent:NN:0, monkey:NN:0, ...	animal:NN:0, species:NN:1, ...	rat::NN:conj_and, white-footed:JJ:amod, ...
mouse:NN:1	keyboard:NN:1, computer:NN:0, printer:NN:0 ...	device:NN:1, equipment:NN:3, ...	click:NN:-prep_of, click:NN:-nn, ...
keyboard:NN:0	piano:NN:1, synthesizer:NN:2, organ:NN:0 ...	instrument:NN:2, device:NN:3, ...	play:VB:-dobj, electric:JJ:amod, ..
keyboard:NN:1	keypad:NN:0, mouse:NN:1, screen:NN:1 ...	device:NN:1, technology:NN:0 ...	computer:NN:nn, qwerty:JJ:amod ...

Raw Text



Text with Proto concept IDs

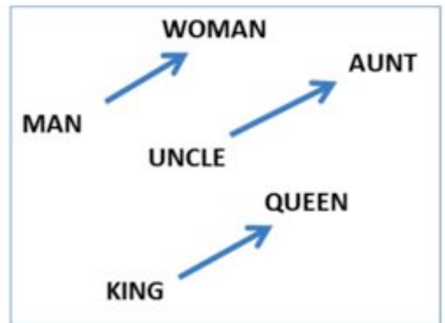
Text with Ontology concept IDs

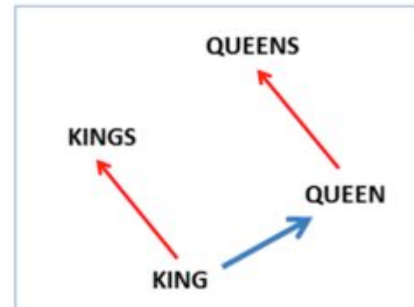
Faralli, S., Panchenko, A., Biemann, C., Ponzetto, S.P. (2016): Linking lexical resources to disambiguated distributional semantic networks. ISWC Resource track 2016, Kobe, Japan

Arithmetic: Word Analogy and Compositionality

VDSMs clearly win here:

- no notion of directionality in a graph
- no notion of arithmetic in a graph





Trust me, I have tried:

- Compositionality in GDSM works for frequently observed combinations but is not generative; unclear how e.g. to yield straightforwardly comparable sentence representations
- king – man + woman = queen   works on a sparse feature representation as well, but computations are cumbersome

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proc. NIPS, pages 3111–3119.

Interpretability and Robustness of Representation

Why are 'anaconda' and 'python' similar?

largest point of critique on dense VDSMs:

- lack of interpretability of dimensions
- when using random sampling methods: re-running the procedure results in different values

because their cosine similarity is 0.95, being most similar in dimensions 54, 3 and 8 while being least similar in dimensions 90, 22 and 15 using random seed 0.

Sparse models:

- readable
- deterministic / reproducible on same corpus
- robust: similar representations on similar corpora

because they share 36 significant syntactic contexts, of which the most salient are:
they coil up, are snakes, swallow, digest, gorge, tighten, and co-occur in conjunctions with other snakes such as rattlesnake, cobra, ..

Interpretable WSID

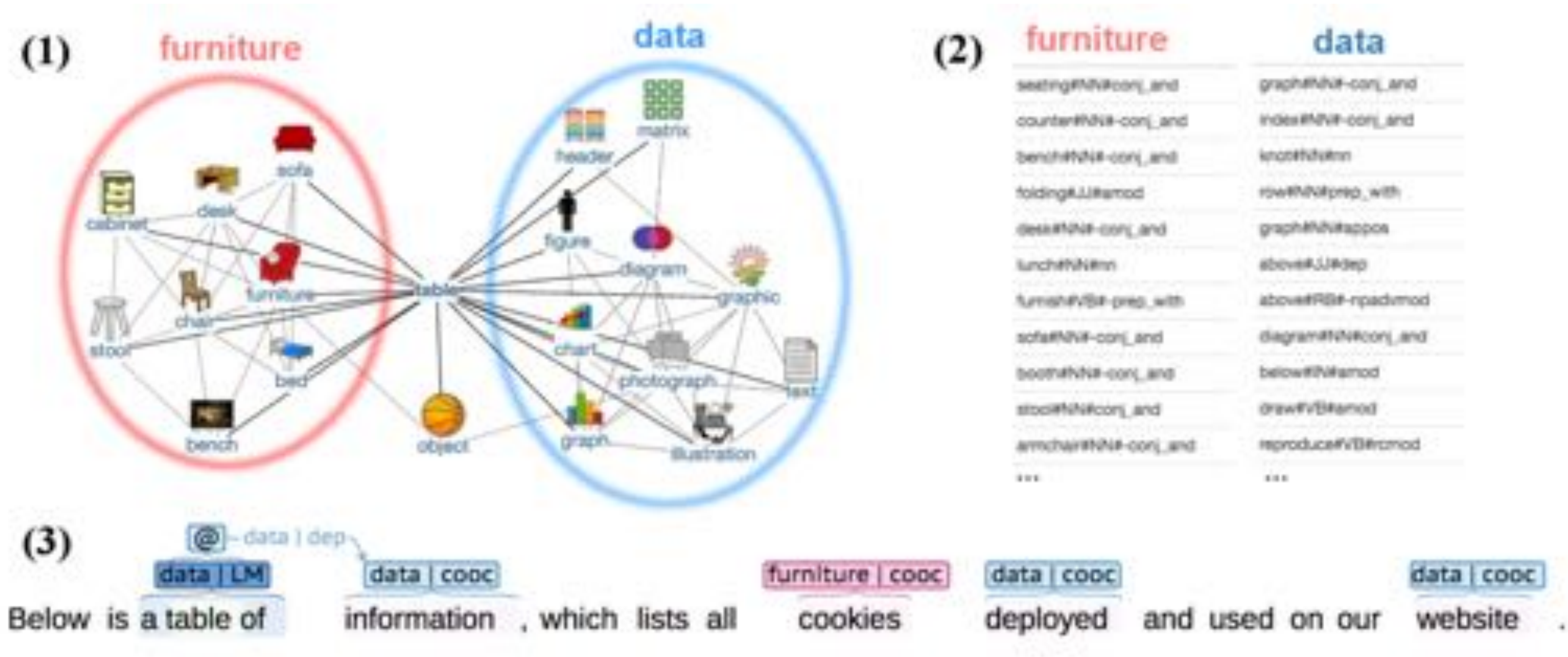


Figure 2: Interpretation of the senses of the word “table” at three levels by our method: (1) word sense inventory; (2) sense feature representation; (3) results of disambiguation in context. The sense labels (“furniture” and “data”) are obtained automatically based on cluster labeling with hypernyms. The “@” sign denotes the target ambiguous word.

Panchenko, A., Ruppert, E., Faralli, S., Ponzetto, S.P., Biemann, C. (2017): Unsupervised Does Not Mean Uninterpretable: The Case for Word Sense Induction and Disambiguation. Proc. EACL 2017, Valencia, Spain

Learnability and Cognitive Plausibility – Anyone?

not well-addressed by neither GDSMs nor VDSMs.

Desired:

- learn continuously and iteratively from a stream of language
  - current models: either batch mode or multiple passes
  - many current models: vocabulary needs to be known beforehand
  - would work with simple counting, but full memorization is not plausible
- cognitive plausibility: represent symbolic reasoning on top of neural brain architecture
  - current models: either symbolic or neural
  - current neural models: per-task, specialized, not whole-brain-ish

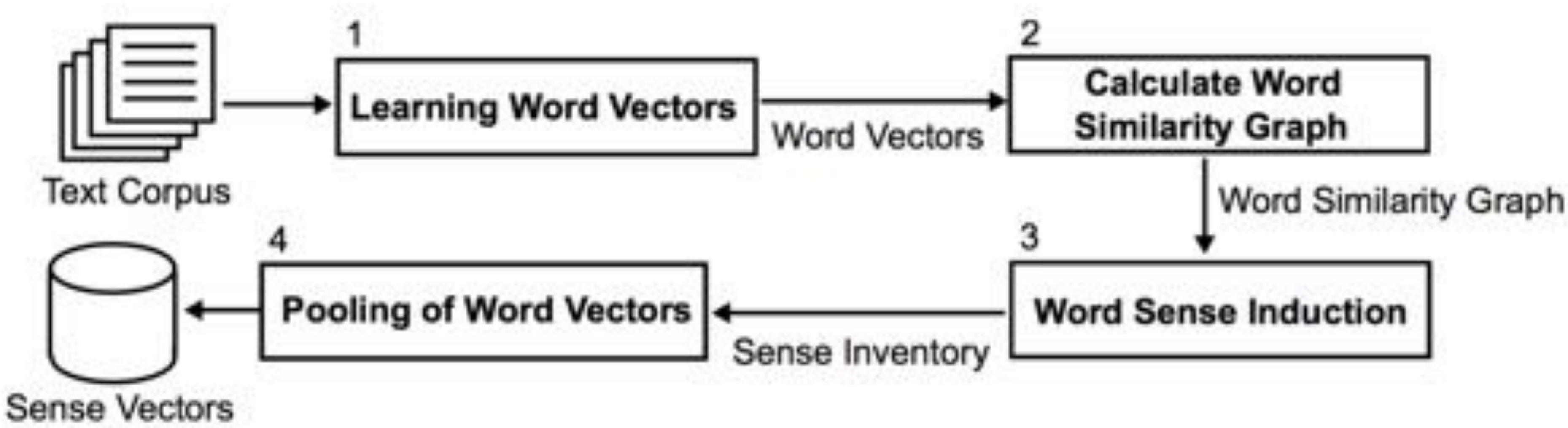
Now, don’t get me wrong ...

- Both representations have their merits!
- Both representations can be retrofitted with mechanisms that overcome their downsides!
- I am not religious – I hope you are not religious, either.

Ways to combine VDSMs and GDSMs:

- modularize steps in your system and use more appropriate representation
- can turn vector spaces into graphs, e.g. along word similarity
- can turn graphs into vector spaces, e.g. by graph embeddings

Example: Word Sense Induction Disambiguation



- Goal of this work: Word Sense Embeddings for ambiguous words for in-context disambiguation
- Use the capability of graph clustering to find the number of senses automatically

Pelevina M., Arefyev N., Biemann C., Panchenko A. (2016) Making Sense of Word Embeddings. In Proceedings of the 1st Workshop on Representation Learning for NLP, Association for Computational Linguistics (ACL). Berlin, Germany [best paper award]

Beyond Vectors and Graphs – so much cool stuff!

- Distributional Relational networks on Knowledge Bases
  http://andrefreitas.org/papers/aaai_distributional_relational_networks_2013.pdf

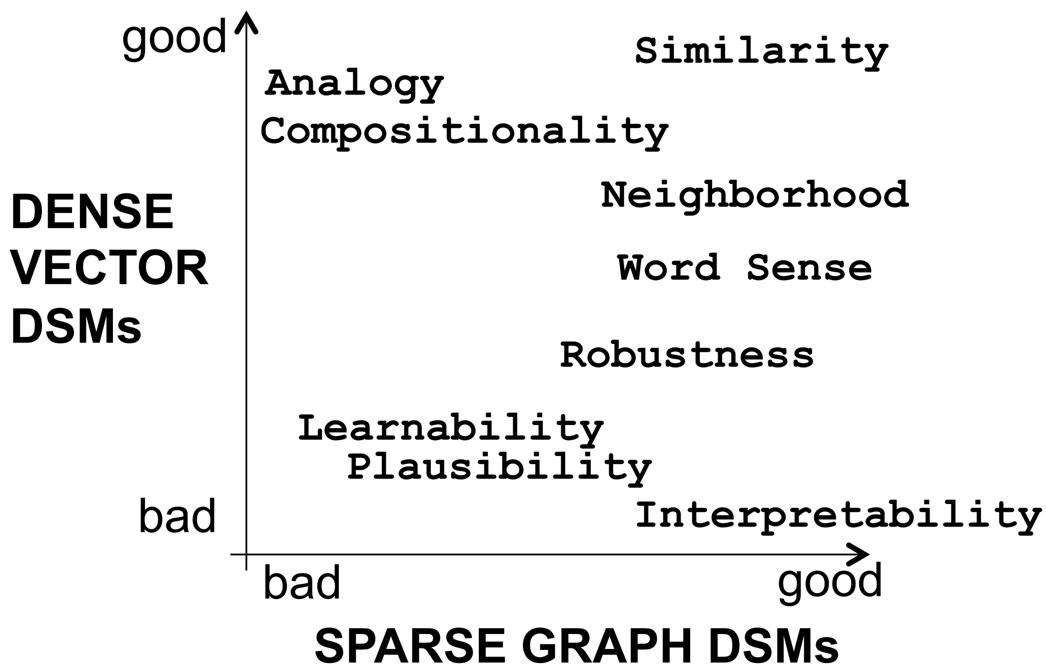
- Multimodal Distributional Models
  https://www.jair.org/media/4135/live-4135-7609-jair.pdf

- Functional Distributional Semantics (with logical forms)
  Combination of Symbolic and Distributional Semantics
  http://www.aclweb.org/anthology/W/W16/W16-1605.pdf

  http://www.cl.cam.ac.uk/~sc609/pubs/aaai07.pdf

Summary

- There are distributional semantic models that are not vector spaces
- Especially, not DENSE vector spaces
- different representations are advantageous for different things
- Choice should depend on the task
- Are you de-biased now?
- at least a little bit?



Thank you ...

... for your

- attention#NN
- scrutiny#NN
- ire#NN
- publicity#NN
- praise#NN
- affection#NN
- enthusiasm#NN
- mind#NN
- wishes#NN
- patience#NN
- wrath#NN
- criticism#NN

and your

- question#NN
- query#NN
- doubt#NN
- concern#NN
- issue#NN
- complaint#NN
- dilemma#NN
- idea#NN
- uncertainty#NN
- matter#NN
- concern#VB
- suggestion#NN

Abstract

Distributional Semantic Models (DSMs) have recently received increased attention, together with the rise of neural architectures for scalable training of dense vector embeddings. While some of the literature even includes terms like ’vectors’ and ’dimensionality’ in the definition of DSMs, there are some good reasons why we should consider alternative formulations of distributional models. As an instance, I present a scalable graph-based solution to distributional semantics. The model belongs to the family of ’count-based’ DSMs, keeps its representation sparse and explicit, and thus fully interpretable. I will highlight some important differences between sparse graph-based and dense vector approaches to DSMs: while dense vector-based models are computationally easier to handle and provide a nice uniform representation that can be compared and combined in many ways, they lack interpretability, provenance and robustness. On the other hand, graph-based sparse models have a more straightforward interpretation, handle sense distinctions more naturally and can straightforwardly be linked to knowledge bases, while lacking the ability to compare arbitrary lexical units and a compositionality operation. Since both representations have their merits, I opt for exploring their combination in the outlook.