Knowledge-lean WSD

(Vector Space Models)

Overview

- **Distributional Semantics with Syntactic Contextualization** *(Thater et al. 2011)*

- **Topic Models for WSD** *(Li et al. 2010)*
  - Latent Dirichlet Allocation - briefly
  - Topic based WSD Models from Li et al.
  - Experimental Setup
  - Results and Conclusion

- **Conclusion**

Distributional Semantics

(Vector Space Models)

Distributional Semantics

(Vector Space Models)

Mechanics

- The ***context surrounding a given word provides information about its meaning***
- Similar words should have similar vectors

Characteristics

- Vector space models are ***simple, wide-coverage, easy to learn***
- Gradual concept of "***semantic similarity***"

Distributional Semantics

(Vector Space Models)

Problems

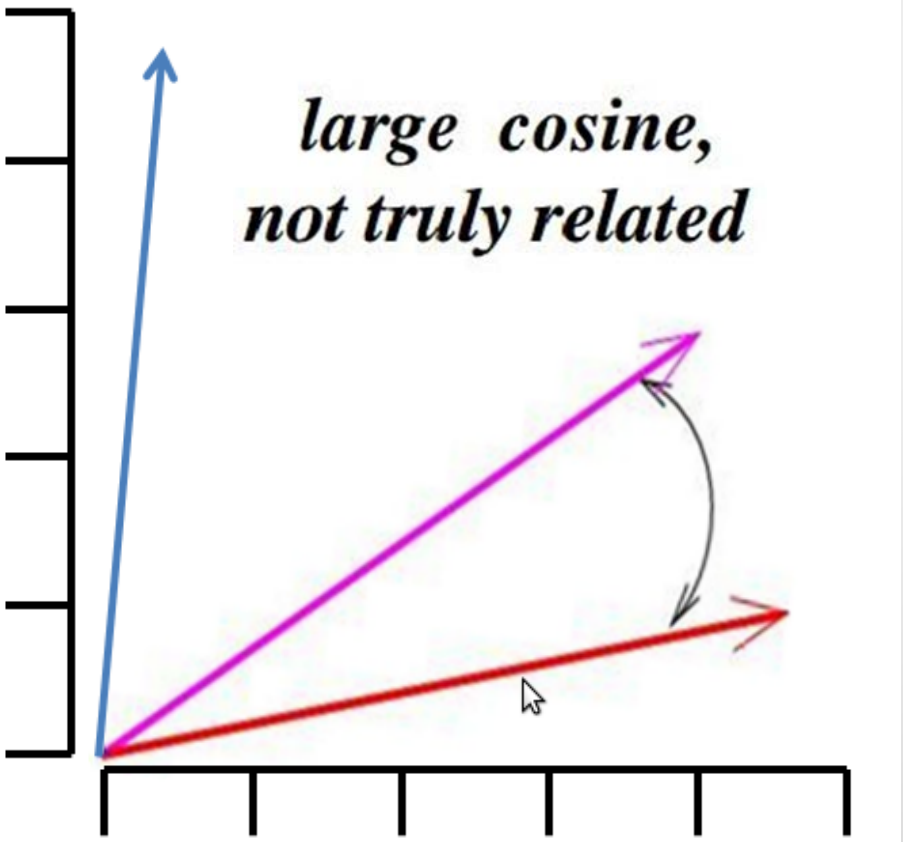
- Vector matrices can be ***sparse and noisy***
- The vector space is very ***ambiguous / insensitive***
  - Vectors for word range over all senses
  - it can only tell us which set of words are similar

auto engine bonnet tyres lorry boot

car emissions hood make model trunk

make hidden Markov model emissions normalize

auto	trunk	Markov	
0	0	0	auto
1	0	0	bonnet
1	0	0	boot
0	1	0	car
0	1	1	emissions
1	0	0	engine
0	0	1	hidden
0	1	0	hood
1	0	0	lorry
0	1	1	make
0	0	1	Markov
0	1	1	model
0	0	1	normalize
0	0	0	tires
0	1	0	trunk
1	0	0	tyres



Distributional Semantics

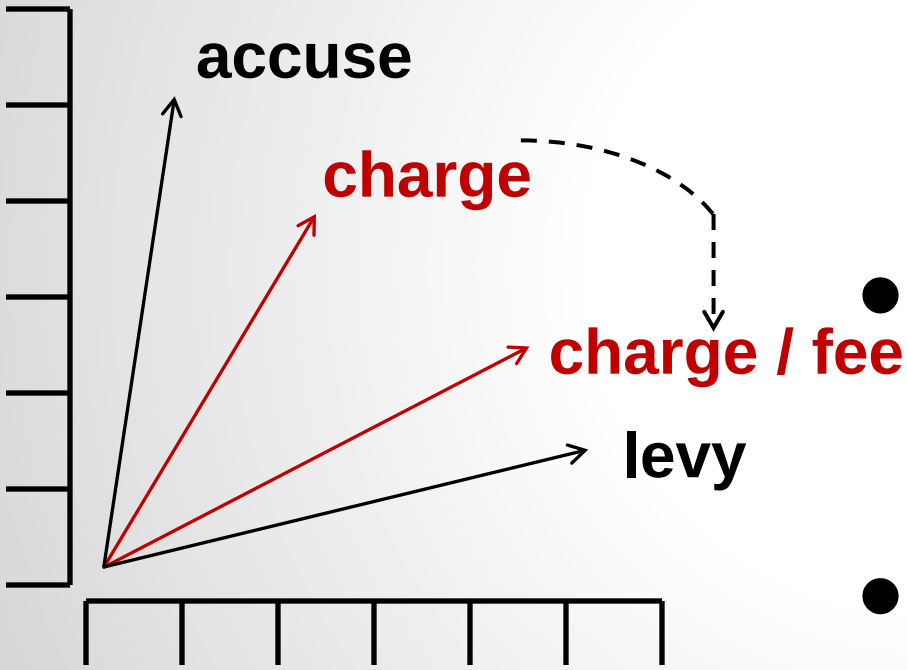
(Vector Space Models)

Solution

- Using specific vectors for ***words in context***
- Reweighting Approach *(Thater et al. 2011)*
  - Reweight vector components with their similarity scores to the context

Distributional Semantics

(Vector Space Models)



- "Fees are **charged** on a pro-rata basis"

  => Levy, require, impose, demand

- "Gilmore was **charged** with possessing a bomb"

  => Indict, accuse, accuse of

- "USB outlet will **charge** without adapter"

  => Recharge, supply electricity, charge up

Distributional Semantics

(Vector Space Models)

- Given an ***occurrence of a word (w) in the context of another (wc)***, *related by the syntactic relation (rc)*, the contextualized vector will be updated as such:

$$\mathbf{v}_{r_c,w_c}(w) := \sum_{r \in R, w' \in W} \alpha_{r_c,w_c,r,w'} \cdot f(w,r,w') \cdot \mathbf{e}_{(r,w')}$$

Quantifies the degree to which a vector dimension (r,w') is compatible with the observed context (rc,wc)

The basic vector that captures the association strength between w and the context word w' in relation r.

$$\mathbf{v}_{r_c,w_c}(w) := \sum_{r \in R, w' \in W} \alpha_{r_c,w_c,r,w'} \cdot f(w,r,w') \cdot \mathbf{e}_{(r,w')}$$

$$PMI(w,r,w') = \log \frac{p(w,w' \mid r)}{p(w,\cdot \mid r)p(\cdot,w' \mid r)}$$

$$\mathbf{v}_{r_c,w_c}(w) := \sum_{r \in R, w' \in W} \alpha_{r_c,w_c,r,w'} \cdot f(w,r,w') \cdot \mathbf{e}_{(r,w')}$$

No contextualization: $\alpha_{r_c,w_c,r,w'} := 1$

In this case the definition of $\mathbf{v}_{r_c,w_c}(w)$ coincides with that of $\mathbf{v}(w)$.

$$\mathbf{v}_{r_c,w_c}(w) := \sum_{r \in R, w' \in W} \alpha_{r_c,w_c,r,w'} \cdot f(w, r, w') \cdot \mathbf{e}_{(r,w')}$$

Strict contextualization:

$$\alpha_{r_c,w_c,r,w'} := \delta_{r_c,r}\delta_{w_c,w'}$$

$$= \begin{cases} 1 & \text{if } r_c = r \text{ and } w_c = w' \\ 0 & \text{else} \end{cases}$$

Here, we only retain the one dimension (r_c, w_c) that is licensed by the context and set all other dimensions to 0.

$$\mathbf{v}_{r_c,w_c}(w) := \sum_{r \in R, w' \in W} \alpha_{r_c,w_c,r,w'} \cdot f(w, r, w') \cdot \mathbf{e}_{(r,w')}$$

Similarity-based contextualization:

$$\alpha_{r_c,w_c,r,w'} := \delta_{r_c,r} \cdot \mathrm{sim}(w_c, w')$$

$$= \begin{cases} \mathrm{sim}(w_c, w') & \text{if } r_c = r \\ 0 & \text{else} \end{cases}$$

Here, we generalize over the surface context and license all words w' that are semantically similar to the context word w_c.

Contextualized Vector Evaluation Task

Basic Idea

- Rank Substitution candidates for words in context

Train and Preprocessing

- Gigaword corpous
- Standford dependency parser

Test

- Lexical Substitution Task dataset (2007)

Contextualized Vector Results

Model	GAP	Random
Erk and Padó (2008)	27.4†	N/A
Erk and Padó (2010)	38.6‡	28.5
Dinu and Lapata (2010)	42.9	30.3
Thater et al. (2010)	46.0	30.0
Our model	51.7	30.0

† Cited from Erk and Padó (2010). The result refers to a small subset of the Lexical Substitution Task dataset.

‡ Evaluated on nouns, verbs, and adjectives (not adv.).

Contextualized Vector WSD Evaluation Task

Basic Idea

- Extract substitution candidates from WordNet (synonyms, hyponyms, hypernyms)
- Predict the synset associated with the most similar substitution candidate (fine grained)

Contextualized Vector WSD Evaluation Task

Coarse-grained (Model –MFS):

1. ***Collect all sense paraphrases*** for each sense of the target word from WordNet

2***. Calculate similarity measure*** between the contextualized vector of the target word to the paraphrase candidate

Contextualized Vector WSD Evaluation Task

Coarse-grained (Model –MFS):

3. ***Normalize the scores*** of the all synsets so that they sum to 1.

4. Compute probabilites for each sense cluster by ***aggregating over its constituent synsets***.

5. Predict that the ***correct sense is the most probable aggregated and nomarlized sysnet***.

Contextualized Vector WSD Evaluation Task

Coarse-grained (Model +MFS):

- Same as Step 1-5 of Model **-MFS** and additionally
- ***Multiply the score of each synset with its prior probability from WordNet sense frequency***
- If system fails to make prediction, fall back to MFS

Contextualized Vector Coarse-grained WSD Results

Model	+MFS	-MFS
Random	52.4	52.4
Most frequent sense (MFS)	78.9	—
Our Model	80.9	78.7†

- Thater et al. 2010 simple contextualized vector model beats MFS baseline, it also beats a complex statistical method

(to be conitnued...)


TO BE
CONTINUED...

Topic Models (LDA)

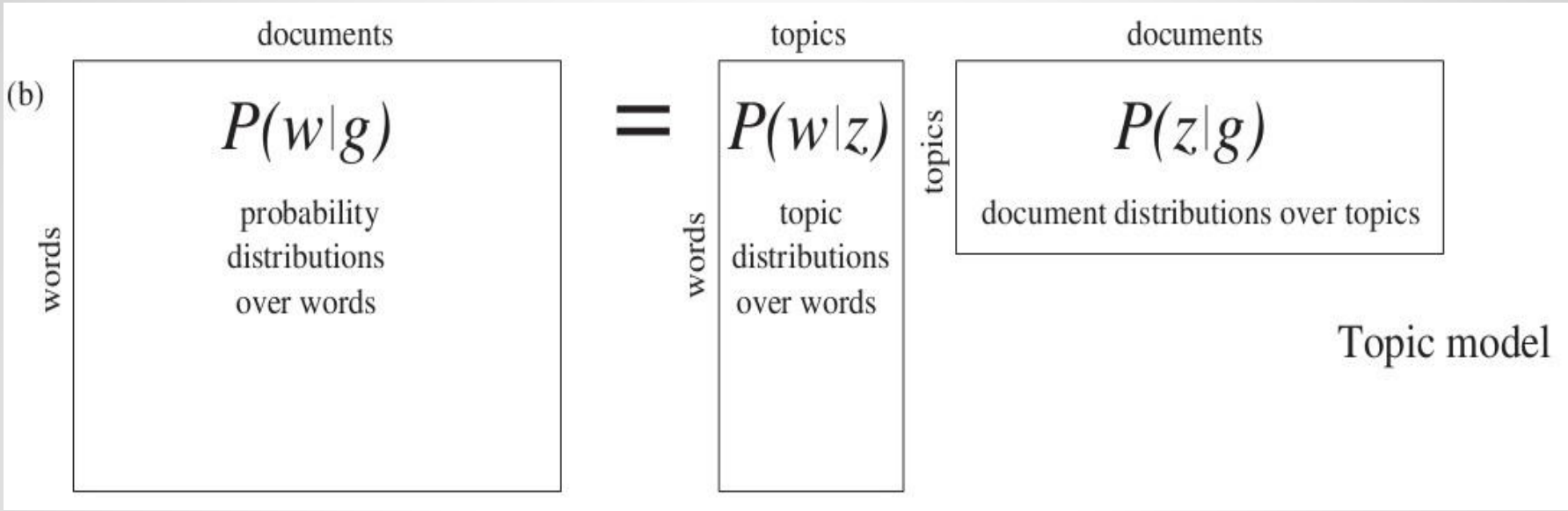
(Briefly – Blei et al. 2003, 2011)

Latent Dirichlet Allocation

- LDA is a ***generative probabilistic model*** *of a corpus*.

- The basic idea is that
  - ***docs are represented as random mixtures over latent topics*** $p(z|doc)$, where
  - ***each topic is characterized by a distribution over words*** $p(word|z)$,

Latent Dirichlet Allocation



- We have the p(w|g) matrix from the corpus.
- The generative model will introduce the topic variable (z) and ***iteratively guess the p(w|z) and p(z|g) matrices to compute p(w|g)***
- until the computed p(w|g) matrix looks like the one from the corpus

Latent Dirichlet Allocation

- LDA is a Bayesian method to "statistically infer" the three probabilities:

$$p(w|d) = \sum_z p(z|d)p(w|z)$$

- For now, we just assume that we can ***"magically"*** get the probability matrices for p(z|d) and p(w|z)
  - Inputs: A large corpus, the α and β parameters
  - Outputs: A list of zs, p(z|d) and p(w|z)

Latent Dirichlet Allocation

- LDA is a Bayesian method to "statistically infer" the three probabilities:

$$p(w|d) = \sum_z p(z|d)p(w|z)$$

LDA requires two Dirichlet parameters and Wang et al. (2009) suggests:

- $\alpha = \frac{50}{\#topics}$
- $\beta = 0.01$

Latent Dirichlet Allocation

- LDA is a Bayesian method to "statistically infer" the three probabilities:

$$p(w|d) = \sum_z p(z|d)p(w|z)$$

- LDA assumes that the ***p(z)*** is uniform,
  - i.e. all topics have equal probability of happening
  - Although this simplifies the task of finding p(sense|context), it is also a very strong assumption.

Sense Disambiguation using Topic Models

(main presentation)

Topic Model WSD (Li et al. 2010)

Approach

- Choosing the best sense ***based on conditional probability of sense paraphrases given a context***
- They implemented 3 models for topic model based WSD.
- Different models can be used ***depending on how much "knowledge" is available***:

Topic Model WSD

- Model 1
  - Requires ***WordNet*** and ***prior distribution of senses***
- Model 2
  - Requires ***WordNet*** and uses ***sense frequency*** to estimate prior distribution of senses
- Model 3
  - **No "knowledge"**, maximizes the sense-context probability by maximizing cosine value of two document vectors

Topic Model WSD

- To assign a correct sense (s) to a target word (w) in the context (c),
- Li et al. (2010) ***maximizes the conditional probabilities of a senses given a context***:

$$s = \arg\max_{s_i} p(s_i|c)$$

- A sense (s_i) is a ***collection of 'paraphrases'*** that capture the partial meaning of the sense.

Topic Model WSD:

(Sense Paraphrases)

Model 1 and Model 2:

Li et al. (2010) obtain the paraphrase from from WordNet 2.1.

- The ***word forms***, ***glosses*** and ***example sentences*** of the synset itself and a set of selected reference synsets.

- The context will be treated as ($c = dc$) and sense paraphrase is ($s = ds$)

Sense Paraphrases (Model I and II)

- The *word forms*, *glosses* and *example sentences* of the synset itself and a set of ***selected reference synsets***

POS	Paraphrase reference synsets
N	hyponyms, instance hyponyms, member holonyms, substance holonyms, part holonyms, member meronyms, part meronyms, substance meronyms, attributes, topic members, region members, usage members, topics, regions, usages
V	Troponyms, entailments, outcomes, phrases, verb groups, topics, regions, usages, sentence frames
A	similar, pertainym, attributes, related, topics, regions, usages
R	pertainyms, topics, regions, usages

Table 1: Selected reference synsets from WordNet that were used for different parts-of-speech to obtain word sense paraphrase. N(noun), V(verb), A(adj), R(adv).

Sense Paraphrases (Model I and II)

For example, the sense paraphrases for **"quickly"** are:

- S: (adv) **quickly**, rapidly, speedily, chop-chop, apace, in short order (with speed) *"he works quickly"; "John got ready in short order"*
  - *pertainym*
    - W: (adj) quick [Related to: quickly] (accomplished rapidly and without delay) *"was quick to make friends"; "his quick reaction prevented an accident"; "hoped for a speedy resolution of the problem"; "a speedy recovery"; "he has a right to a speedy trial"*
    - W: (adj) rapid [Related to: rapidly] (done or occurring in a brief period of time) *"a rapid rise through the ranks"*
    - W: (adj) speedy [Related to: speedily] (accomplished rapidly and without delay) *"was quick to make friends"; "his quick reaction prevented an accident"; "hoped for a speedy resolution of the problem"; "a speedy recovery"; "he has a right to a speedy trial"*
  - *antonym*
- S: (adv) promptly, **quickly**, quick (with little or no delay) *"the rescue squad arrived promptly"; "come here, quick!"*
  - *pertainym*
    - W: (adj) prompt [Related to: promptly] (ready and willing or quick to act) *"she is always prompt to help her friends"*
    - W: (adj) quick [Related to: quickly] (accomplished rapidly and without delay) *"was quick to make friends"; "his quick reaction prevented an accident"; "hoped for a speedy resolution of the problem"; "a speedy recovery"; "he has a right to a speedy trial"*
- S: (adv) cursorily, **quickly** (without taking pains) *"he looked cursorily through the magazine"*
  - ***pertainym***
    - W: (adj) cursory [Related to: cursorily] (hasty and without attention to detail; not thorough) *"a casual (or cursory) inspection failed to reveal the house's structural flaws"; "a passing glance"; "perfunctory courtesy"; "In his paper, he showed a very superficial understanding of psychoanalytic theory"*
    - W: (adj) quick [Related to: quickly] (hurried and brief) *"paid a flying visit"; "took a flying glance at the book"; "a quick inspection"; "a fast visit"*

Sense Paraphrases (Model I and II)

The sense paraphrases for **"quickly"** are:

"quickly, rapidly, speedily, chop-chop, apace, in short order, quick, rapid, speedy, quick (with little or no delay), prompt, cursorily, quickly, cursory, ...

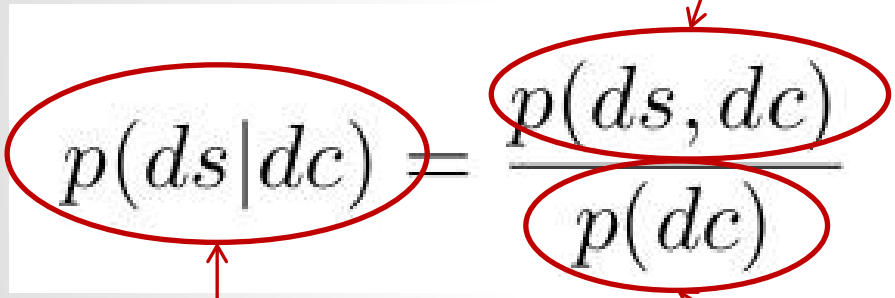
(and also the glosses and example sentences, they are too long to list here)"

Model 1

- Model I directly ***maximizes the conditional probability of the sense given the context***, i.e. $p(ds|dc)$
  - where the sense is modeled as a ***paraphrase document (ds)*** and the context as a ***context document (dc)***.
- The conditional probability of sense given context:

$$p(ds|dc) = \frac{p(ds, dc)}{p(dc)}$$

Joint probability a sense paraphrase occurs with the particular context



This tells us how probable a sense paraphrase is given the context

This is the probability that a particular context will occur. (Normalizing factor)

Model 1

- When we introduce the ***latent topic variable (z)*** through generative process and p(ds,dc) rewrites as such:

$$p(ds, dc) = \sum_{z} p(ds)p(z|ds)p(dc|z)$$

First we introduce the topic (z), so p(ds,dc) equals ot the sum of p(ds,dc,z) across all topics:

$$p(ds, dc) = \sum p(ds, dc, z)$$

apply Bayes rule:

$$p(ds, dc, z) = P(ds, dc|z)p(z)$$

$$p(ds, dc) = \sum P(ds, dc|z)p(z)$$

Then we assume that given (z), a paraphrase document (ds) is generated independently of the context document (dc):

$$p(ds, dc) = \sum p(ds|z)p(dc|z)p(z)$$

$$p(ds, dc) = \sum p(ds|z)p(dc|z)p(z)$$

Apply Bayes rule again on p(ds|z),

$$p(ds|z) = \frac{p(z|ds)p(ds)}{p(z)}$$

$$p(ds, dc) = \sum \frac{p(z|ds)p(ds)p(dc|z)\cancel{p(z)}}{\cancel{p(z)}}$$

$$p(ds, dc) = \sum_{z} p(ds)p(z|ds)p(dc|z)$$

The conditional probability of sense given context:

$$p(ds|dc) = \frac{p(ds, dc)}{p(dc)}$$

$$p(ds, dc) = \sum_z p(ds)p(z|ds)p(dc|z)$$

$$p(ds|dc) = \frac{\sum p(ds)p(z|ds)p(dc|z)}{p(dc)}$$

$$p(ds|dc) = \frac{\sum p(ds)p(z|ds)p(dc|z)}{p(dc)}$$

Apply Bayes rule to p(dc|z):

$$p(dc|z) = \frac{p(z|dc)p(dc)}{p(z)}$$

$$p(ds|dc) = \frac{\sum p(ds)p(z|ds)p(z|d)\cancel{p(dc)}}{\cancel{p(dc)}p(z)}$$

$$p(ds|dc) = p(ds)\sum_z \frac{p(z|dc)p(z|ds)}{p(z)}$$

$$p(ds|dc) = p(ds) \sum_z \frac{p(z|dc)p(z|ds)}{p(z)}$$

Since p(z) is a uniform distribution according to uniform Dirichlet priors, so:

$$p(ds|dc) \propto p(ds) \sum_z p(z|dc)p(z|ds)$$

Model I

- Model I directly ***maximizes the conditional probability of the sense given the context***,

$$p(ds|dc) \propto p(ds) \sum_z p(z|dc)p(z|ds)$$

- Model I:

$$\arg\max_{ds_i} p(ds_i) \sum_z p(z|dc)p(z|ds_i)$$

Model I

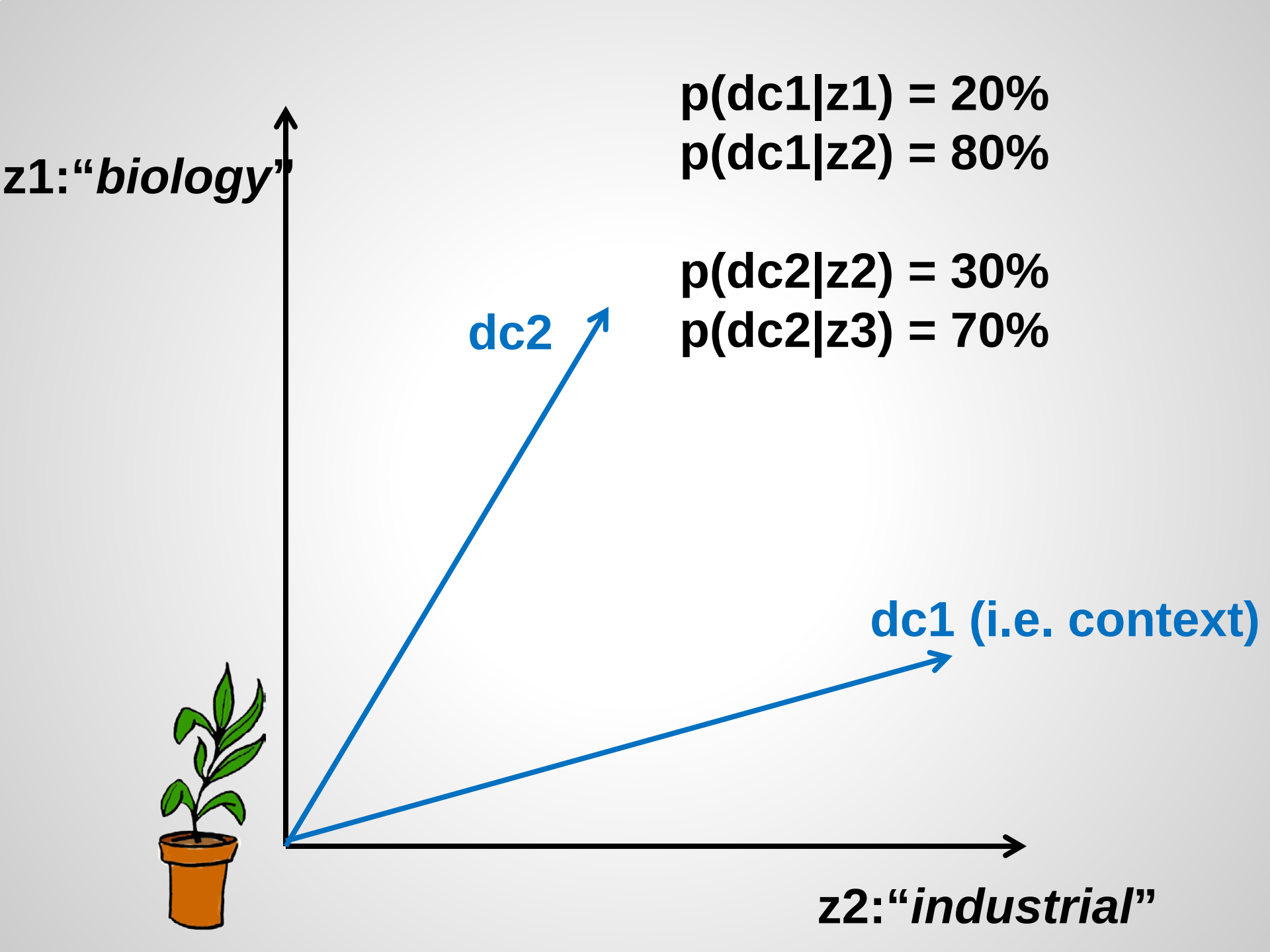
Disadvantages

- Prior sense distribution $p(ds)$ is not always available.
- Assuming uniform $p(ds)$ is also not feasible because sense distribution is often highly skewed (McCarthy, 2009)
- Hence Model I is still knowledge dependent

Model 2

- **Model 2** bypass the ***p(ds)*** by maximizing the cosine value of two document vectors that encode document-topic frequencies, ***v(z|dc) and v(z|ds)***

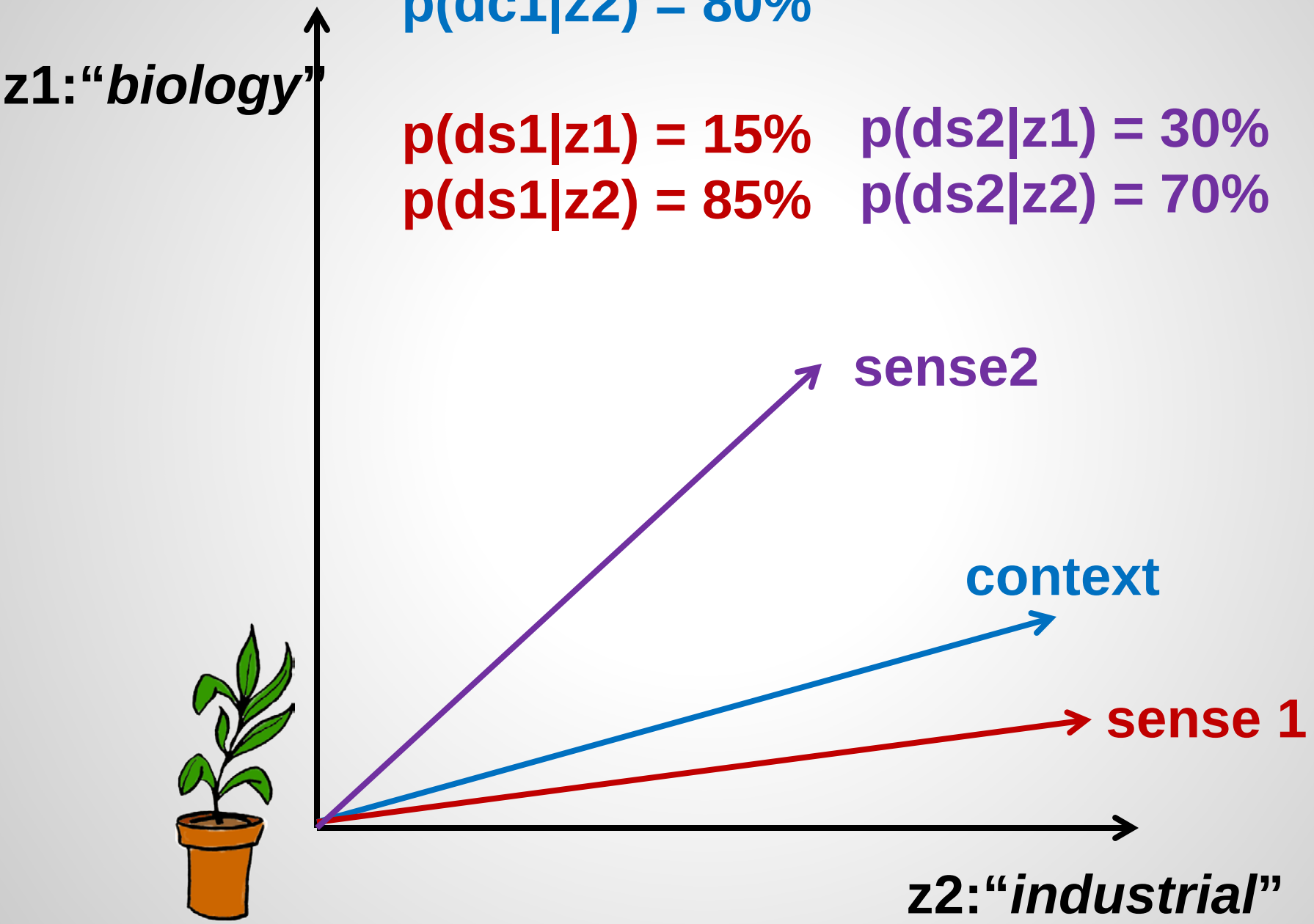
$$\arg\max_{ds_i} \cos(v(z|dc), v(z|ds_i))$$

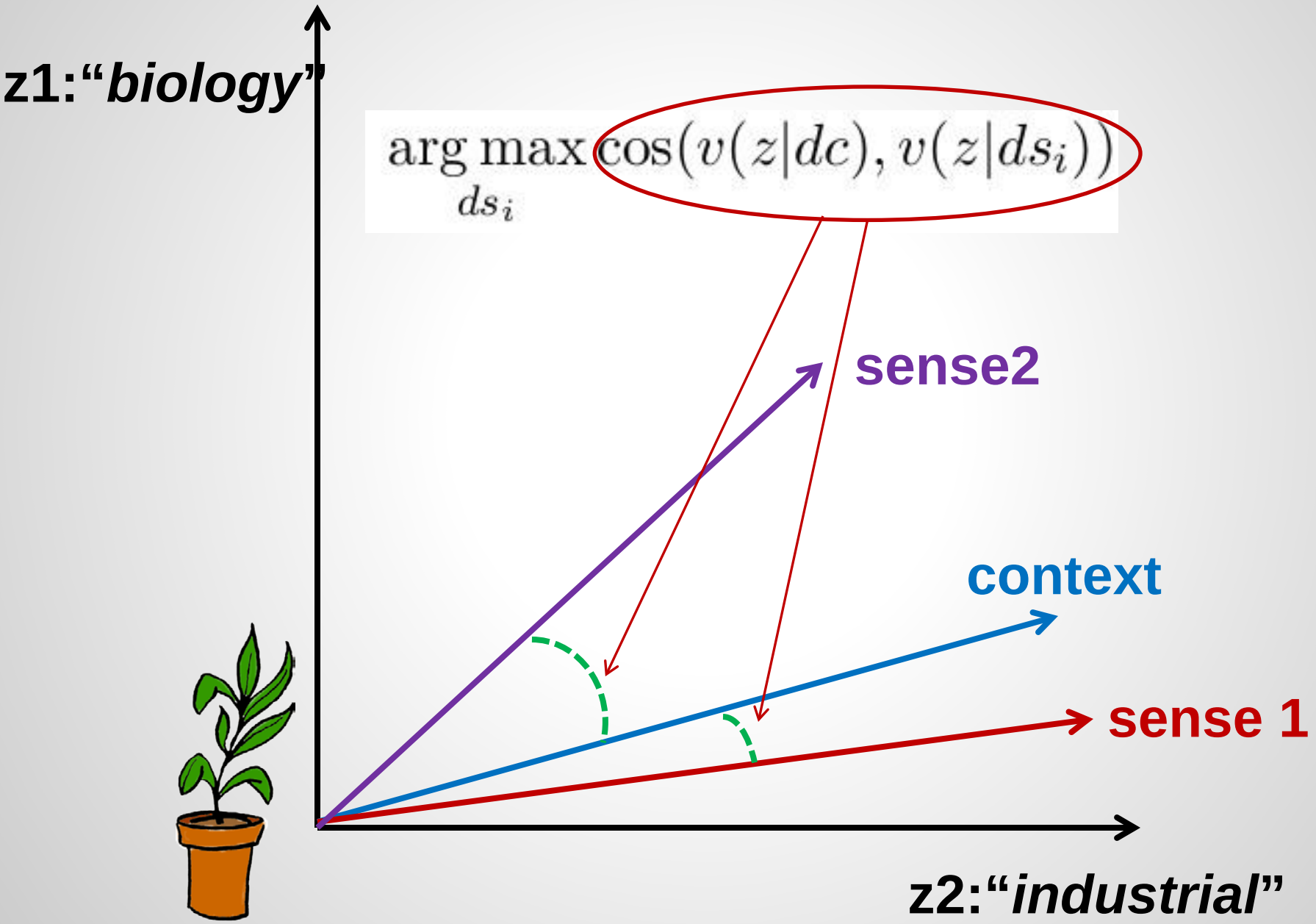

p(dc1|z1) = 20%
p(dc1|z2) = 80%
p(dc2|z2) = 30%
p(dc2|z3) = 70%
z1:“biology”
dc2
dc1 (i.e. context)
z2:“industrial”

p(dc1|z1) = 20%
p(dc1|z2) = 80%

p(ds1|z1) = 15%
p(ds1|z2) = 85%

p(ds2|z1) = 30%
p(ds2|z2) = 70%




z1:“biology”
$\arg\max_{ds_i} \cos(v(z|dc), v(z|ds_i))$
sense2
context
sense 1
z2:“industrial”

Model 3

- Model 2 is still knowledge dependent, it requires WordNet
- Hence Model 3 proposes a fully unsupervised model, where:
  - the ***sequence of independent words*** are treated sense paraphrases (***ds***)
  - the ***contexts*** are treated as documents (***dc***)

$$\arg\max_{qs_i} \max_{w_i \in qs} \sum_{z} p(w_i|z)p(z|dc)$$

Why take argmax instead of product of all conditionaly probabilities?

Because

- Taking product will ***penalize long paraphrases***
- Want to avoid modelling the generation of specific paraphrases
  - The point of the model is to ***induce the topic***, not to find the most similar paraphrases
  - e.g. "Rock the boat" = "break the norm" | "cause trouble“, the topic of ***"rock the boat" can be represented by "norm" and "trouble"*** instead of introducing noise from "break" and "cause“

Experiments and Results

(Li et al. 2010)

Experimental Setup (Data)

Topic Distribution Inference (for all models):

- English Wikipedia Dump (2009-07-13)
  - ESA implementation
  - Snowball stopword filter

Sense Paraphrases (for model 1+2):

- WordNet 2.1

Instance Context:

- 5 diff. context settings [±1w, ±5, ±10, current sentence, whole text]

Coarse grain WSD task (Semeval-2007)

- 5377 words in 5 articles, 3 from WSJ, 1 from Wiki, 1 from Italian painters biographies
- 1108 nouns, 591 verbs, 362 adjectives, 208 adverbs
- annotated with sense clusters from WordNet 2.1

Coarse-grain WSD

System	Noun	Verb	Adj	Adv	All
UoR-SSI	84.12	78.34	85.36	88.46	83.21
NUS-PT	82.31	78.51	85.64	89.42	82.50
UPV-WSD	79.33	72.76	84.53	81.52	**78.63***
TKB-UO	70.76	62.61	78.73	74.04	**70.21′**
MII–ref	78.16	70.39	79.56	81.25	76.64
MII+ref	80.05	70.73	82.04	82.21	**78.14′**
MI+ref	79.96	75.47	83.98	86.06	**79.99***
BL_{mfs}	77.44	75.30	84.25	87.50	**78.99***

Results (Coarse Grained)

- Comparing TKB-UO (70.21%) to the unsupervised MIII (70.21%), significantly better
- Comparing UPV-WSD (78.63%) to supervised MI (79.9%), no significance but still better.
- MI also better than MFS (78.99%)

Context Analysis

Context	Ate.	Pre.	Rec.	F1
$\pm 1w$	91.67	75.05	68.80	71.79
$\pm 5w$	99.29	77.14	76.60	76.87
$\pm 10w$	100	77.92	77.92	77.92
text	100	76.86	76.86	76.86
sent.	100	**78.14**	**78.14**	**78.14**

Table 3: Model II performance on different context size. attempted rate (Ate.), precision (Pre.), recall (Rec.), F-score (F1).

Context Analysis

- Using smaller context reduces both precision and recall
  - because small context causes > all-zero topic assignment for documents only containing words that are not in the vocabulary
- Using whole text does not perform well
  - possibly because using full text folds in too much noise

Fine grain WSD task (Semeval-2007)

- 3500 words in 3 articles from WSJ, 465 lemmas
- 159 nouns, 296 verbs, 10 untagged
- annotated with sense clusters from WordNet 2.1

Fine-grain WSD

- MI+ref beats MFS and best system in SemEval-2007 (nice game)

- Fine-grained WSD is a nice game but most ***industrial-grade MT systems uses coarse-grain WSD or lexical substitution system***

Fine-grain WSD

System	F-score
RACAI	52.7 ±4.5
BL_{mfs}	55.91±4.5
MI+ref	**56.99**±4.5

Table 4: Model performance (F-score) for the fine-grained word sense disambiguation task.

Potentially ambiguous expression (Sporleder and Li, 2009)

- 3964 instances of 17 potentially ambiguous English idioms
- manually annotated as *literal* or *nonliteral*

Literal vs Non-literal WSD

System	Prec_l	Rec_l	F_l	Acc.
Base_{maj}	-	-	-	78.25
co-graph	50.04	69.72	58.26	78.38
boot.	71.86	66.36	69.00	87.03
Model III	67.05	81.07	73.40	87.24

Table 5: Performance on the literal or nonliteral sense disambiguation task on idioms. literal precision (Prec_l), literal recall (Rec_l), literal F-score (F_l), accuracy(Acc.).

Literal vs Non-literal WSD

- MIII performs the best and ***it outperforms semi-supervised bootstrapping***

- The task is sensitive to quality of paraphrases

PREVIOUSLY . . .

Coarse-grained WSD (Li et al. 2010)

System	Noun	Verb	Adj	Adv	All
UoR-SSI	84.12	78.34	85.36	88.46	83.21
NUS-PT	82.31	78.51	85.64	89.42	82.50
UPV-WSD	79.33	72.76	84.53	81.52	**78.63***
TKB-UO	70.76	62.61	78.73	74.04	**70.21**′
MII–ref	78.16	70.39	79.56	81.25	76.64
MII+ref	80.05	70.73	82.04	82.21	**78.14**′
MI+ref	79.96	75.47	83.98	86.06	**79.99***
BL_{mfs}	77.44	75.30	84.25	87.50	**78.99***

Contextualized Vector Coarse-grained WSD Results

Model	+MFS	-MFS
Random	52.4	52.4
Most frequent sense (MFS)	78.9	—
Our Model	80.9	78.7†

- Thater et al. 2010 simple contextualized vector model beats MFS baseline, it also beats a complex statistical method

(to be conitnued...)

Conclusion

Model	+MFS	-MFS
Random	52.4	52.4
Most frequent sense (MFS)	78.9	—
Li et al. (2010)	81.3‡	78.8‡
Our Model	80.9	78.7†

- Thater et al. (2010) **simple contextualized vector model** beats Li et al. (2010) **probabilistic topic model**
- But they can be combined...

Conclusion

- “The **diners** at my table simply lit more Gauloises [...],”
  - ***Thater et al.*** model correctly predicts the sense ***“person eating a meal” of the target diners, based on the leading sense paraphrase eater.***
  - ***Li et al.*** system predicts the sense ***“passenger car where food is served”***, which fits the general topic similarly well, but is highly implausible in the given syntactic context.

Conclusion

- “The program text, or source, was converted into machine instructions using a special program called a **compiler**,”
  - o ***Thater et al.*** system ranks the sense paraphrase author over program and thus incorrectly ***predicts the sense “person who compiles encyclopedias***.”
  - o ***Li et al.*** system is able to leverage topical clues to correctly ***predict the software sense of compiler***,

Conclusion

Model	+MFS	-MFS
Random	52.4	52.4
Most frequent sense (MFS)	78.9	—
Li et al. (2010)	81.3‡	78.8‡
Our Model	80.9	78.7†
Combined system	82.2	78.9

- Combine the systems by
  - Average their predicted probability distributions
  - Fallback to Li et al. for instances not covered by our model
- Improvement of 0.9% is statistically significant ($p < 0.01$)

Fin.

Summary

- **Contextualized Vector Model** *(Thater et al. 2010)*
  - A vector model with syntactic contextualization
  - Using ***reweighted vectors components with their similarity scores to the context*** for lexical substitution
  - ***Mapping lexical substitution*** candidates to WordNet senses for coarse-grain WSD
  - *If sense distribution is known,* ***multiply the score of each synset with its prior probability from WordNet*** *sense frequency*

Summary

- **Topic Models for WSD** *(Li et al. 2010)*
  - Uses either context word or WordNet synsets as sense paraphrase
  - Choosing the best sense ***based on conditional probability of sense paraphrases given a context***
  - If available, ***uses sense distribution*** or WordNet sense frequencies ***as prior weight***