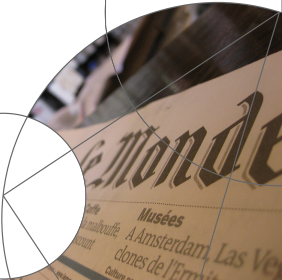
CNRS - INPT - UPS - UT1 - UTM

A Supervised Distributional Approach for Free Paraphrasing of Noun Compounds

Tim Van de Cruys
Stergos Afantenos
Philippe Muller

Toulouse Research Institute for Computer Science

14 June 2013



Introduction

- Our approach: combination of unsupervised distributional word space model with supervised classification
- Step 1: extract feature representation from unsupervised distributional word space model
- Step 2: use feature representation to train supervised maximum entropy classifier

Step 1: Word space model

- Word space models are based on the DISTRIBUTIONAL HYPOTHESIS
  - Take a word and its contexts:
    - tasty *sooluceps*
    - sweet *sooluceps*
    - stale *sooluceps*
    - freshly baked *sooluceps*
- By looking at a word's context, one can infer its meaning

Step 1: Word space model

- Word space models are based on the DISTRIBUTIONAL HYPOTHESIS

  - Take a word and its contexts:
    - tasty *sooluceps*
    - sweet *sooluceps*
    - stale *sooluceps*
    - freshly baked *sooluceps*

  $\Rightarrow$ **food**

- By looking at a word's context, one can infer its meaning

Step 1: Word space model

- Word space models are based on the DISTRIBUTIONAL HYPOTHESIS

  - Take a word and its contexts:
    - tasty *sooluceps*
    - sweet *sooluceps*
    - stale *sooluceps*
    - freshly baked *sooluceps*

- By looking at a word's context, one can infer its meaning

Matrix

- captures co-occurrence frequencies of two entities

	red	tasty	fast	second-hand
raspberry	2	1	0	0
strawberry	2	2	0	0
car	1	0	1	2
truck	1	0	1	1
flavour	0	2	0	0

Matrix

- captures co-occurrence frequencies of two entities

	red	tasty	fast	second-hand
raspberry	7	9	0	0
strawberry	12	6	0	0
car	7	0	8	4
truck	2	0	3	4
flavour	0	10	0	0

Matrix

- captures co-occurrence frequencies of two entities

	red	tasty	fast	second-hand
raspberry	56	98	0	0
strawberry	44	34	0	0
car	23	0	31	39
truck	4	0	18	29
flavour	0	66	0	0

Matrix

- captures co-occurrence frequencies of two entities

	red	tasty	fast	second-hand
raspberry	728	592	1	0
strawberry	1035	437	0	2
car	392	0	487	370
truck	104	0	393	293
flavour	0	389	0	0

Matrix

- captures co-occurrence frequencies of two entities

	red	tasty	fast	second-hand
raspberry	728	592	1	0
strawberry	1035	437	0	2
car	392	0	487	370
truck	104	0	393	293
flavour	0	389	0	0

- weighted using pointwise mutual information

Compositional representation

- Paraphrases are determined by semantics of both head noun and modifier noun
- We want to compute a joint, compositional representation of head noun with modifier noun
- Simple vector-based multiplicative model (Mitchell & Lapata 2008)
- $p_i = u_i v_i$ for each feature i

Compositional representation

- Paraphrases are determined by semantics of both head noun and modifier noun
- We want to compute a joint, compositional representation of head noun with modifier noun
- Simple vector-based multiplicative model (Mitchell & Lapata 2008)
- $p_i = u_i v_i$ for each feature i

	red	tasty	fast	second-hand
raspberry	728	592	1	0
flavour	0	389	0	0

Compositional representation

- Paraphrases are determined by semantics of both head noun and modifier noun
- We want to compute a joint, compositional representation of head noun with modifier noun
- Simple vector-based multiplicative model (Mitchell & Lapata 2008)
- $p_i = u_i v_i$ for each feature i

	red	tasty	fast	second-hand
raspberry flavour	0	230288	0	0

Step 2: maximum entropy classifier

We estimate a supervised probability distribution:

$$h : \mathcal{X} \mapsto \mathcal{Y}$$

- $\mathcal{X}$: training vectors with non-zero features from word space model
- $\mathcal{Y}$: paraphrase label

Estimating a supervised probability distribution

$$P(y|x) = \frac{1}{Z(y)} exp\left(\sum_{i=1}^{m} w_i f_i(x, y)\right)$$

- $f_i(x, y)$ are the features extracted from word space model
- w_i is a weight associated with each feature obtained by maximizing the log-likelihood of the training data with respect to the model
- $Z(y)$ is a normalization factor
- Results exceeding threshold ϕ are considered as paraphrases
- $\phi = 0.01$, calculated on 20% held out data on the training set

Set of paraphrase labels

We substitute the nouns that appear in the training set's paraphrases by dummy variables; example:

compound	paraphrase	paraphrase label
textile company	company that makes textiles	Y that makes Xs
textile company	company that produces textiles	Y that produces Xs
textile company	company in textile industry	Y in X industry

- paraphrase label needs to appear with at least two different compounds
- set of 285 possible paraphrases
- potential drawback: loss of recall

Implementational details

- Frequency co-occurrence information extracted from UKWAC corpus
- Word space model
    - 5K nouns by 2K co-occurring context words (most frequent, excluding stop words)
    - window of 5 words to left and right
    - make sure all nouns from train and test set are included
- algorithms implemented in Python, maxent classifier uses maximum entropy modeling toolkit

Experiments

Three baselines:

- assign the two most frequent paraphrases to each instance
  - *Y of X*, *Y for X*
- assign the four most frequent paraphrases
  - *Y of X*, *Y for X*, *Y on X*, *Y in X*
- assign the 285 paraphrase labels collected on the training set

Parameter settings:

- two thresholds on the probability of the paraphrase label
  - very low threshold (0.001) $\rightarrow$ more paraphrases
  - low threshold (0.01) $\rightarrow$ less paraphrases

Results

model	ϕ	isomorphic	non-isomorphic
baseline (2)	–	.058	**.808**
baseline (4)	–	.090	.633
baseline (all)	–	**.332**	.200
multiplicative	.01	.130	.548
	.001	.270	.259
head noun	.01	.136	.536
	.001	.277	.302

Discussion

- Baselines models achieve respectable scores
  - conservative baseline (few paraphrases) gets high non-isomorphic score
  - liberal baseline (lots of paraphrases) gets high isomorphic score
- same tendency for our model scores:
  - higher threshold $\rightarrow$ fewer paraphrases $\rightarrow$ high non-isomorphic score
  - lower threshold $\rightarrow$ more paraphrases $\rightarrow$ more balanced scores
- no significant differences between both models: head noun model and multiplicative model reach similar results

Examples

house price ($\theta = .01$)	
multiplicative	**head noun**
price of house	price of house
price for house	price for house
price paid for house	price for purchasing house
price for purchasing house	price paid for house
price given for house	price given for house

Examples

bank governor ($\theta = .01$)	
multiplicative	**head noun**
governor at bank	governor of bank
governor in bank	governor for bank
governor for bank	governor in bank
governor made in bank	governor in charge of bank
governor kept in bank	governor responsible for bank
governor of money in bank	governor who controls bank
governor in charge of bank	governor in bank department
governor of bank	governor appointed for bank
governor made by bank	governor at bank
governor from bank	

Examples

forest officer ($\theta = .01$)	
multiplicative	**head noun**
officer where forest is done	officer in forest department
officer of forest	officer responsible for forest
officer that fights forest	officer in charge of forest
officer to forest	officer for forest
officer for foresting	officer who controls forest
officer with forest	officer engaged in forest
officer of forests	officer of forest
officer who controls forest	officer in forest
officer in forest department	officer appointed for forest
officer from forest	
officer meant for forest	
officer on forest	
officer provided for forest	
officer in charge of forest	
officer by forest	

Conclusion

- system for paraphrasing that combines unsupervised word space model and supervised classifier
- free paraphrasing, but only for paraphrases present in training set
- no significant difference between multiplicative model and head noun model
- balance with regard to the two metrics directly depends on the number of proposed paraphrases