Distributional Semantics: From Ad Hoc Solutions to Persistent Models

Marco Baroni

Center for Mind/Brain Sciences (CIMeC)
University of Trento

IS-LTC 2008
Ljubljana, 16/10/2008

Collaborators and related work

- Most empirical work reported here done in collaboration with Alessandro Lenci (University of Pisa)
- Other collaborations on relevant topics:
  - Brian Murphy, Massimo Poesio, Eduard Barbu, Roberto Zamparelli (University of Trento)
  - Raffaella Bernardi (University of Bolzano)
  - Stefan Evert (University of Osnabrück)
- Similar ideas are currently being explored (at least) by:
  - Peter Turney (NRC of Canada)
  - Katrin Erk (University of Texas at Austin) and Sebastian Padó (Stanford University)

In short

- Corpus-based distributional methods have been amazingly successful in tackling semantics-related tasks
- However, different semantic challenges have remained largely disconnected
  - Each class of tasks requires specifying a statistical model, going back to the corpus, collecting the statistics, and running the trained model
- This is not how semantics works in humans (and not a very practical way to have it working in machines)
- Can we develop a unified corpus-based semantic model, that can be tuned to the various tasks?
- Yes

Distributional semantics

- What are text corpora good for?

Distributional semantics

- What are text corpora good for?
- Somewhat counterintuitively, to find out about word meaning

Distributional semantics

- What are text corpora good for?
- Somewhat counterintuitively, to find out about word meaning
- Just look at any recent issue of Computational Linguistics, proceedings of ACL, etc.

Distributional semantics

- “You should tell a word by the company it keeps” (Firth, 1957)
- “[T]he semantic properties of a lexical item are fully reflected in appropriate aspects of the relations it contracts with actual and potential contexts [...] [T]here are are good reasons for a principled limitation to linguistic contexts” (Cruse, 1986)
- Structuralism on wheels: (a lot about) the meaning of a linguistic unit (morpheme, word, phrase, construction. . .) is given by (or can be inferred from) network of contextual relations it entertains in text collections (corpora)

Distributional semantics

He put the bagva on the stove and waited for the water to boil

A big, ugly bagva had been following us for the last three miles

Distributional semantics

- Approach popular mainly for its empirical successes
  - The more semantic tasks we can solve with corpus-based evidence, the more plausible it is that we are tackling into something genuinely semantic
- Semantically relevant, purely linguistic contexts might play an important role in child concept acquisition (Beals, 1997)
- Corpus-derived co-occurrence statistics predict whole-brain neural activity in response to conceptual stimuli (Mitchell et al., 2008)

Outline

Classic semantic tasks

- Finding taxonomically similar words/concepts
- Finding word/concept pairs instantiating similar relations
- Constraining word/concept composition

Classic semantic tasks

- Finding taxonomically similar words/concepts
- Finding word/concept pairs instantiating similar relations
- Constraining word/concept composition
- Not discussed here:
  - Semantics at the sentence, paragraph, document levels
    - Topic models, gist extraction, paraphrasing, textual entailment, etc.
  - Semantic shifts in context
    - Word sense disambiguation, anaphora resolution, co-composition phenomena

Finding taxonomically similar words

Lund and Burgess, 1998, Landauer et al. 1998, Schütze 1997, Sahlgren 2006

- Meaning of words defined by *set of contexts* in which word occurs in running text
- Similarity of words represented as *geometric distance* among *context vectors*
  - (Alternatively: similarity of probability distributions, relative entropy. . .)
- Impressive results in tasks such as synonym identification, concept categorization

Finding pairs instantiating the same relation

- Two ways to approach the problem:
  - Relation extraction (Hearst, 1992, Cimiano and Wenderoth, 2005, Pantel and Pennacchiotti, 2006, ...)
    - Pick a relation (e.g., hyponymy, function), and look for pairs that instantiate it (*dog-animal*, *airplane-transportation*)
  - Relational similarity (Turney 2006)
    - Pick a pair, and find other pairs instantiating the same relation (*dog-animal*: *airplane-vehicle*, *banana-fruit*, ...)

Finding pairs instantiating the same relation

- Two ways to approach the problem:
  - Relation extraction (Hearst, 1992, Cimiano and Wenderoth, 2005, Pantel and Pennacchiotti, 2006, . . .)
    - Pick a relation (e.g., hyponymy, function), and look for pairs that instantiate it (*dog-animal*, *airplane-transportation*)
  - Relational similarity (Turney 2006)
    - Pick a pair, and find other pairs instantiating the same relation (*dog-animal*: *airplane-vehicle*, *banana-fruit*, . . .)
- Many tasks not explicitly coached in these terms can be framed as relational tasks
  - E.g., find relation linking compound elements (Girju et al., 2007)
    - *brick house* vs. *doll house*

Finding pairs instantiating the same relation

- Typically, target word pairs represented by vector of co-occurrences with corpus-extracted *links* connecting them
- E.g., *dog-animal* and *airplane-vehicle* might share links such as: *such as*, *and other*, *is a kind of*
- Relation extraction (as in, e.g., Pantel and Pennacchiotti, 2006): similarity to set of manually coded pairs instantiating target relation
- Relational similarity: similarity to single pair

Finding pairs instantiating the same relation

- Typically, target word pairs represented by vector of co-occurrences with corpus-extracted *links* connecting them
- E.g., *dog-animal* and *airplane-vehicle* might share links such as: *such as*, *and other*, *is a kind of*
- Relation extraction (as in, e.g., Pantel and Pennacchiotti, 2006): similarity to set of manually coded pairs instantiating target relation
- Relational similarity: similarity to single pair
- Promising (although generally not outstanding) results in harvesting pairs linked by a specific relation and modeling analogical intuitions

Measuring composition plausibility

- Imposing constraints on compositionality by assessing the acceptability of verb-noun, adjective-noun, noun-noun combinations
- A classic instantiation of this task: modeling selectional preferences (Resnik, 1996, Rooth et al., 1999, Erk, 2007, Padó et al., 2007)
  - *boil a potato* vs. *boil an idea*
- Important, since it taps into *productivity* of semantic system (Fodor and Pylyshyn, 1988)
  - *boil a topinambur* vs. *boil an idea*

Measuring composition plausibility

- In recent corpus-based approach (Padó et al., 2007), this is simply a special case of taxonomic similarity
  - Construct a vector of the "prototypical" argument of X (e.g., object of *boil*), by averaging over vectors of words that often fill this slot
  - Measure similarity of Y (e.g., *topinambur* and *idea*) to prototypical vector: good arguments should be more similar to prototype than bad arguments

Measuring composition plausibility

- In recent corpus-based approach (Padó et al., 2007), this is simply a special case of taxonomic similarity
  - Construct a vector of the "prototypical" argument of X (e.g., object of *boil*), by averaging over vectors of words that often fill this slot
  - Measure similarity of Y (e.g., *topinambur* and *idea*) to prototypical vector: good arguments should be more similar to prototype than bad arguments
- Promising results in tasks simulating human acceptability ratings

Measuring composition plausibility

- In recent corpus-based approach (Padó et al., 2007), this is simply a special case of taxonomic similarity
  - Construct a vector of the "prototypical" argument of X (e.g., object of *boil*), by averaging over vectors of words that often fill this slot
  - Measure similarity of Y (e.g., *topinambur* and *idea*) to prototypical vector: good arguments should be more similar to prototype than bad arguments
- Promising results in tasks simulating human acceptability ratings
- Many apparently unrelated tasks can be modeled with analogous techniques
  - E.g., the "logical metonymy" problem (Pustejovsky, 1994; Lapata and Lascarides, 2003)
    - *enjoy (reading) the book* vs. *enjoy (eating) the ice-cream*
    - Look for verbs whose object prototype has high cosine with noun under investigation

Many tasks, many models

- The standard paradigm:
  - Identify and operationalize task
  - Design algorithm to tackle the task, and specify needed resources to be extracted from corpus
  - Extract relevant statistics from corpus
  - Evaluate

Many tasks, many models

- The standard paradigm:
  - Identify and operationalize task
  - Design algorithm to tackle the task, and specify needed resources to be extracted from corpus
  - Extract relevant statistics from corpus
  - Evaluate
- Compare to traditional theoretical semantics (e.g., Montague Grammar), where first a model is developed, and then various tasks are identified and used to test it (possibly updating model in response to empirical issues)

Many tasks, many models

- The standard paradigm:
  - Identify and operationalize task
  - Design algorithm to tackle the task, and specify needed resources to be extracted from corpus
  - Extract relevant statistics from corpus
  - Evaluate
- Compare to traditional theoretical semantics (e.g., Montague Grammar), where first a model is developed, and then various tasks are identified and used to test it (possibly updating model in response to empirical issues)
- Can we do the same with corpus-based computational semantics?
- (More on *why* we might want to do it later)

Outline

General framework

See also Padó and Lapata (2007)

- A semantic model is a set of tuples:
  `arg1 link arg2 weight`
- Model is trained from corpus data once for all tasks
- Task adaptation by set of rules to construct co-occurrence matrix from tuples

Training

- Any standard technique to collect weighted tuples
- E.g., from output of dependency parser, using lexico-syntactic patterns, etc.
- Weighting by co-occurrence frequency, log-likelihood ratio, MI, entropy, etc.

The trained model

arg1	link	arg2	weight
banana	for	eating	23.9
kill	obj	man	25.9
banana	is	yellow	25.7
yellow	is^{-1}	banana	22.2
dog	subj^{-1}	bark	32.3
kill time	subj	student	10.3
sing	tense	perfect	15.4

Task adaptation

- Row/column splitting
- Select row items (targets)
- Select column items (contexts)
- Build prototypes, if needed
- Measure similarity

Task adaptation
Row/column splitting

`arg1`		by	`link`	`arg2`
`arg1`	`arg2`	by	`link`	
`arg1`	`link`	by	`arg2`	

(Other cases can be reconducted to these by adding tuples with swapped `arg1`/`arg2`, inverted `link`, etc.)

Task adaptation

Select targets (row items) and contexts (column items)

- Target selection:
  - If the task involves, e.g., comparing nouns, no need to include verb or adjective rows in the matrix
  - More often: pick only the nouns of interest, etc.
- Context selection:
  - In part trivial
    - E.g., add only dimensions with non-0 values for at least some of targets
  - In part interesting
    - E.g., different similarity space if only “functional” `link arg2` pairs are used as dimensions

Task adaptation
Build prototypes

- Generalization by measuring similarity to abstract prototypes
- E.g., decide if *to boil a topinambur* is semantically acceptable by measuring distance of *topinambur* from prototypical object of *to boil*

Task adaptation
Build prototypes

- Generalization by measuring similarity to abstract prototypes
- E.g., decide if *to boil a topinambur* is semantically acceptable by measuring distance of *topinambur* from prototypical object of *to boil*
- In practice, prototype construction involves:
  - Selecting rows that will contribute to the prototype (e.g., nouns that have positive weight with `obj boil` in a tuple)
  - Averaging dimensions of selected rows
- Lots of interesting technical issues I will skip:
  - How to select the contributing rows (external or internal criteria, or a mix)
  - How to average (sum, multiply, assign different weights, etc.)

Measuring similarity

- Again, lots of interesting technical issues, to be skipped
- In all experimental results below, similarity measured by cosine distance between target rows in co-occurrence matrix

Outline

Theoretical possibilities and empirical evaluation

- Show that all row-column splits lead to meaningful tasks
- Show that classic semantic tasks are covered
- Some empirical evidence that a single model (in the sense we just defined) can reach state-of-the-art results in classic and new tasks

TupleWare beta

- A simple starting point to explore how far we can go with the "single model, many tasks" idea
- Training corpus: ukWaC, 2.25 billion tokens from the Web (Ferraresi et al, 2008), pre-parsed with Lin's MINIPAR
- `arg1`, `arg2`
  - Top 20k most frequent nouns, top 5k most frequent verbs in corpus
- `link`
  - Top 30 verb-to-noun dependency relations in the corpus and their inverse (e.g., `pound with hammer` and `hammer` $\texttt{with}^{-1}$ `pound`)
  - Impoverished model: only noun co-occurrence information used to represent verbs, verb co-occurrence for nouns
- Weight: local MI (Evert 2004) of `arg1 link arg2` tuples

`arg1` by `link` `arg2`

	with leash	subj^{-1} walk	subj^{-1} run	has owner	obj^{-1} pet	subj^{-1} 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
car	0	0	1	3	0	0

`arg1` by `link` `arg2`

- Similarity among single word/concept vectors
  $\longrightarrow$ Taxonomic similarity tasks
- Similarity to prototypes
  $\longrightarrow$ Compositionality tasks

TupleWare output

Nearest neighbours of hammer among set of concrete objects

neighbour	*cosine*
knife	0.43
screwdriver	0.27
chisel	0.23
scissors	0.22
pencil	0.22
spoon	0.19
pen	0.18
bottle	0.08
truck	0.07
telephone	0.05

TupleWare output

Nearest neighbours of cook among selection of verbs

neighbour	*cosine*
roast	0.56
cook	0.44
grill	0.31
fry	0.20
boil	0.17
steam	0.10
jog	0.02
chat	0.02
speak	0.01
swim	0.01

Nominal concept categorization
44 concrete concepts

- 24 natural concepts
  - 15 animals: 7 birds, 8 ground animals
  - 9 vegetables: 4 fruits, 5 greens
- 20 artifacts
  - 13 tools
  - 7 vehicles
- Comparison with state-of-the-art algorithms in unsupervised clustering task

Results

Percentage *purity* of clusters

model	*6 categories*	*3 categories*	*2 categories*
TupleWare	75	84	100
DV	73	89	95
SVD	79	75	59

Similar results for Italian, and for verb categorization task

Correlation with Rubenstein and Goodenough's (1965) similarity ratings

- Average ratings from 51 subjects for 65 noun pairs on 0-4 scale, e.g.:

car	automobile	3.9
food	fruit	2.7
cord	smile	0.0

Correlation with Rubenstein and Goodenough's (1965) similarity ratings

- Average ratings from 51 subjects for 65 noun pairs on 0-4 scale, e.g.:

car	automobile	3.9
food	fruit	2.7
cord	smile	0.0

- Pearson correlations coefficients (in percentage):

TupleWare	69
Best DV (from Padó and Lapata, 2007)	62
Best DV (cosine-based)	47
SVD	56

Compositionality

Predict/quantify selectional restrictions

- Select 50 nouns from highest weight tuples including `link verb` pair under investigation (e.g., 50 nouns most associated with $\mathtt{obj}^{-1}$ `kill`)
- Build prototype by summing (normalized) vectors representing these nouns in `link verb` space
- Felicity of unseen noun as filler of `link verb` slot (e.g., as an object of killing) measured as cosine of noun vector to prototype
- NB: since we are interested in generalization, if target noun happens to be in top 50 set, we subtract its vector from prototype before computing cosine

Compositionality

Acceptability of some potential objects of kill

object	*cosine*
kangaroo	0.51
person	0.45
robot	0.15
hate	0.11
flower	0.11
stone	0.05
fun	0.05
book	0.04
conversation	0.03
sympathy	0.01

Compositionality

Acceptability of some potential instruments of kill

object	*cosine*
hammer	0.26
stone	0.25
brick	0.18
smile	0.15
flower	0.12
antibiotic	0.12
person	0.12
heroin	0.12
kindness	0.07
graduation	0.04

Compositionality

Correlation with human acceptability judgments of Padó 2007

- 211 noun-verb pairs rated on 7-point scale with noun in subject and object position ($\sim$20 raters per pair)
  - How common is it for an ear to hear something?
  - How common is it for an ear to be heard?

Compositionality

Correlation with human acceptability judgments of Padó 2007

- 211 noun-verb pairs rated on 7-point scale with noun in subject and object position (∼20 raters per pair)
  - How common is it for an ear to hear something?
  - How common is it for an ear to be heard?

model	*percentage coverage*	*percentage correlation*
TupleWare	92	52
Padó et al. (2007)	98	48
Padó et al. (2006)	97	51

Compositionality

Correlation with human acceptability judgments of Padó 2007

- 211 noun-verb pairs rated on 7-point scale with noun in subject and object position (∼20 raters per pair)
  - How common is it for an ear to hear something?
  - How common is it for an ear to be heard?

model	*percentage coverage*	*percentage correlation*
TupleWare	92	52
Padó et al. (2007)	98	48
Padó et al. (2006)	97	51

- To study next: how does composition affect semantic representation in context? (Mitchell and Lapata, 2008; Erk and Padó, 2008)
  - John attended the school
  - John entered the school

arg1 arg2 by link

		$subj^{-1}$	obj^{-1}	$with^{-1}$	$like^{-1}$
dog	bark	5	0	0	5
cat	meow	4	0	0	3
motorcycle	roar	2	0	0	1
motorcycle	ride	0	5	2	2

arg1 arg2 by link

- Similarity among specific vectors
  $\longrightarrow$ Relational similarity
- Similarity to prototypes
  $\longrightarrow$ Relation extraction

`arg1 arg2` by `link`

- Similarity among specific vectors
  $\longrightarrow$ Relational similarity
- Similarity to prototypes
  $\longrightarrow$ Relation extraction
- No formal evaluation of TupleWare, yet
  - In Baroni et al. (submitted), we show that a similar approach to relation extraction outperforms the state-of-the-art Espresso method

Some near neighbours of *hammer-pound*

noun	*verb*	*cosine*
knife	slice	0.91
screwdriver	remove	0.90
scissors	chop	0.90
chisel	remove	0.91
peacock	display	0.90
pen	mark	0.89
turtle	draw	0.89
lion	eat	0.87
rocket	hit	0.87
helicopter	transport	0.87

Some near neighbours of *hammer-wield*

noun	*verb*	*cosine*
helicopter	pilot	1.00
pear	pick	1.00
cow	slaughter	0.99
turtle	kill	0.99
screwdriver	insert	0.99
mushroom	warm	0.99
telephone	use	0.99
corn	ground	0.99
bottle	refill	0.99
truck	use	0.99

arg1 link by arg2

		murderer	victim	drug	house	enemy	book
kill	obj	2	9	0	0	7	0
kill	subj	9	1	8	0	7	1
die	subj	3	9	0	0	6	0
burn	obj	2	5	2	7	5	7
burn	subj	2	6	2	5	3	6

Similarity of “verb slots”

- Less studied, but a useful task, that might bring us closer to topics of interest to formal semanticists
- E.g., find out that the object of killing is similar to the subject of dying
  - Long tradition of meaning decomposition approaches to verbal semantics (Dowty, Jackendoff. . .):
    kill(x,y) as *cause(x,die(y))*
- E.g., find out that the patient of boiling can also surface as its (inchoative) subject, whereas this does not happen with mincing (Levin and others):
  - the cook melts the cheese → the cheese melts
  - the cook minces the cheese → *the cheese minces

Predicting the causative/inchoative alternation

- For alternating verbs `verb obj`, should be similar to `verb subj-intr` (the same things that are melted melt)
- For non-alternating verbs, the two slots should not be similar (the things that are minced tend to be different from those that mince them)

Predicting the causative/inchoative alternation

- TupleWare tested on:
  - 232 alternating causatives/inchoatives (*melt*)
  - 170 non-alternating transitives (*mince)*

Predicting the causative/inchoative alternation

- TupleWare tested on:
  - 232 alternating causatives/inchoatives (*melt*)
  - 170 non-alternating transitives (*mince)*
- Median cosine between `verb obj` and `verb subj-intr`:
  - Within alternating class: 0.35
  - Within non-alternating class: 0.16
- Difference in cosines highly significant

Predicting the causative/inchoative alternation

- TupleWare tested on:
  - 232 alternating causatives/inchoatives (*melt*)
  - 170 non-alternating transitives (*mince)*
- Median cosine between `verb obj` and `verb subj-intr`:
  - Within alternating class: 0.35
  - Within non-alternating class: 0.16
- Difference in cosines highly significant
- Also, cosine between `verb obj` and `verb subj-intr` significantly higher than cosine between `verb obj` and `verb subj-tr` for alternating, but not for non-alternating
  - the cook melts the cheese; the cheese melts
  - the cook minces the cheese; the cook minces

Other tasks in the `arg1 link` by `arg2` space

- Might lead to better verb categorization (categorizing verb slots makes more sense than categorizing verbs)
  - `murder subj` is more similar to `kill subj`
  - `murder obj` is more similar to `die subj`
- Prototypes in this space might find verb slots that instantiate a certain (proto-semantic) role
  - Agent slots, patient slots, instrument slots, etc.

Outline

Summary

- I proposed that many semantic tasks proposed in the literature can be performed by a model based on storing weighted `arg1 link arg2` tuples, and constructing task-adapted co-occurrence matrices on demand
- Different ways to construct the matrix lead to different but meaningful semantic tasks
- A simple model based on this approach fares reasonably well in a number of distinct standard and non-standard tasks

A slide about storage and efficiency

- Tuples are a lot more compact than matrices, since 0-cells are not stored
  - Further savings from implicit representation of inverse links (no need to store `arg2` `link`$^{-1}$ `arg1`)
- Processing costs to build matrices on demand
  - But notice that in real life you rarely need to build a full matrix
    - E.g., in anaphora resolution you might need to decide among 10 antecedents, not 10,000

Why should we be trying this?

- Humans must do semantics this way: they don’t go back to the training corpus for each new task they encounter
  - Are there interesting limitations emerging from single model constraints? Are they mirrored in human behaviour?

Why should we be trying this?

- Humans must do semantics this way: they don't go back to the training corpus for each new task they encounter
  - Are there interesting limitations emerging from single model constraints? Are they mirrored in human behaviour?
- Agents and robots embedding semantic knowledge should also do it this way

Why should we be trying this?

- Humans must do semantics this way: they don't go back to the training corpus for each new task they encounter
  - Are there interesting limitations emerging from single model constraints? Are they mirrored in human behaviour?
- Agents and robots embedding semantic knowledge should also do it this way
- You should not reinvent the distributional semantics wheel with each new project
  - For its many successes, distributional semantics has produced very few usable resources

Why should we be trying this?

- Humans must do semantics this way: they don't go back to the training corpus for each new task they encounter
  - Are there interesting limitations emerging from single model constraints? Are they mirrored in human behaviour?
- Agents and robots embedding semantic knowledge should also do it this way
- You should not reinvent the distributional semantics wheel with each new project
  - For its many successes, distributional semantics has produced very few usable resources
- Shift focus from extracting statistics from corpus to what you actually want to do with the model
  - "Evolving" the model itself (use model-internal inferences to populate it)
  - Adapting the model to real life tasks
  - A new view on supervised and semi-supervised learning
    - Supervised learning as task adaptation
    - Train classifiers not on corpus data, but on model statistics

Some references

M. Baroni, E. Barbu, B. Murphy and M. Poesio (submitted). StruDEL: A distributional semantic model based on properties and types.

K. Erk and S. Padó (2008). A structured vector space model for word meaning in context. *Proceedings of EMNLP*.

J. Mitchell and M. Lapata (2008). Vector-based models of semantic composition. *Proceedings of ACL*.

S. Padó and M. Lapata (2007). Dependency-based construction of semantic space models. *Computational Linguistics 33:2*, 161-199.

S. Padó, U. Padó and K. Erk (2007). Flexible, corpus-based modelling of human plausibility judgements. *Proceedings of EMNLP*.

P. Turney (2008). A uniform approach to analogies, synonyms, antonyms, and associations. *Proceedings of COLING*.