Lecture 8: Distributional semantics

Models

Getting distributions from text

Real distributions

Similarity

Distributions and classic lexical semantic relationships

`http://xkcd.com/739/`

Plan for this lecture

- Brief introduction to distributional semantics
- Emphasis on empirical findings and relationship to classical lexical semantics (and formal semantics if there's time next lecture)
- See also notes for lecture 8 of Paula Buttery's course: 'Formal models of language'

Introduction to the distributional hypothesis

- Distributional hypothesis: word meaning can be represented by the contexts in which the word occurs.
- Part of a general approach to linguistics that tried to have verifiable notion of concept like ‘noun’ via possible contexts: e.g., occurs after *the* etc, etc
- First experiments on distributional semantics in 1960s, rediscovered multiple times.
- Now an important component in deep learning for NLP (a form of ‘embedding’ — next lecture).

Distributional semantics

Distributional semantics: family of techniques for representing word meaning based on (linguistic) contexts of use.

it was authentic scrumpy, rather sharp and very strong

we could taste a famous local product — scrumpy

spending hours in the pub drinking scrumpy

- Use linguistic context to represent word and phrase meaning (partially).
- Meaning space with dimensions corresponding to elements in the context (features).
- Most computational techniques use vectors, or more generally tensors: aka semantic space models, vector space models.

Distributional semantics

Distributional semantics: family of techniques for representing word meaning based on (linguistic) contexts of use.

it was authentic scrumpy, rather sharp and very strong

we could taste a famous local product — scrumpy

spending hours in the pub drinking scrumpy

- Use linguistic context to represent word and phrase meaning (partially).
- Meaning space with dimensions corresponding to elements in the context (features).
- Most computational techniques use vectors, or more generally tensors: aka semantic space models, vector space models.

Distributional semantics

Distributional semantics: family of techniques for representing word meaning based on (linguistic) contexts of use.

it was authentic scrumpy, rather sharp and very strong

we could taste a famous local product — scrumpy

spending hours in the pub drinking scrumpy

- Use linguistic context to represent word and phrase meaning (partially).
- Meaning space with dimensions corresponding to elements in the context (features).
- Most computational techniques use vectors, or more generally tensors: aka semantic space models, vector space models.

Distributional semantics

Distributional semantics: family of techniques for representing word meaning based on (linguistic) contexts of use.

it was authentic scrumpy, rather sharp and very strong

we could taste a famous local product — scrumpy

spending hours in the pub drinking scrumpy

- Use linguistic context to represent word and phrase meaning (partially).
- Meaning space with dimensions corresponding to elements in the context (features).
- Most computational techniques use vectors, or more generally tensors: aka semantic space models, vector space models.

Outline.

Models

Getting distributions from text

Real distributions

Similarity

Distributions and classic lexical semantic relationships

The general intuition

- **Distributions** are vectors in a multidimensional semantic space, that is, objects with a magnitude (length) and a direction.
- The **semantic space** has dimensions which correspond to possible contexts.
- For our purposes, a distribution can be seen as a point in that space (the vector being defined with respect to the origin of that space).
- *scrumpy* [...pub 0.8, drink 0.7, strong 0.4, joke 0.2, mansion 0.02, zebra 0.1...]
- partial: also perceptual information etc

Contexts 1

Word windows (unfiltered): *n* words on either side of the lexical item.
Example: n=2 (5 words window):

> *| The prime **minister** acknowledged the | question.*

minister [the 2, prime 1, acknowledged 1, question 0]

Contexts 2

Word windows (filtered): *n* words on either side removing some words (e.g. function words, some very frequent content words). Stop-list or by POS-tag.
Example: n=2 (5 words window), stop-list:

> *| The prime **minister** acknowledged the | question.*

minister [prime 1, acknowledged 1, question 0]

Contexts 3

Lexeme window (filtered or unfiltered); as above but using stems.
Example: n=2 (5 words window), stop-list:

| *The prime **minister** acknowledged the | question.*

minister [prime 1, acknowledge 1, question 0]

Contexts 4

Dependencies: syntactic or semantic (directed links between heads and dependents). Context for a lexical item is the dependency structure it belongs to (various definitions).
Example:

> *The prime **minister** acknowledged the question.*

minister [prime_a 1, acknowledge_v+question_n 1]

Parsed vs unparsed data: examples

word (unparsed)	**word (parsed)**
meaning_n	or_c+phrase_n
derive_v	and_c+phrase_n
dictionary_n	syllable_n+of_p
pronounce_v	play_n+on_p
phrase_n	etymology_n+of_p
latin_j	portmanteau_n+of_p
ipa_n	and_c+deed_n
verb_n	meaning_n+of_p
mean_v	from_p+language_n
hebrew_n	pron_rel_+utter_v
usage_n	for_p+word_n
literally_r	in_p+sentence_n

Context weighting

- Binary model: if context *c* co-occurs with word *w*, value of vector $\vec{w}$ for dimension *c* is 1, 0 otherwise.

  ... [a long long long ***example*** *for a distributional semantics] model... (n=4)*

  ... {a 1} {dog 0} {long 1} {sell 0} {semantics 1}...

- Basic frequency model: the value of vector $\vec{w}$ for dimension *c* is the number of times that *c* co-occurs with *w*.

  ... [a long long long ***example*** *for a distributional semantics] model... (n=4)*

  ... {a 2} {dog 0} {long 3} {sell 0} {semantics 1}...

Characteristic model

- Weights given to the vector components express how *characteristic* a given context is for word *w*.
- Pointwise Mutual Information (PMI), with or without discounting factor.

$$pmi_{wc} = log(\frac{f_{wc} * f_{total}}{f_w * f_c})$$

f_{wc}: frequency of word *w* in context *c*
f_w: frequency of word *w* in all contexts
f_c: frequency of context *c*
f_{total}: total frequency of all contexts

Context weighting

- PMI was originally used for finding collocations: distributions as collections of collocations.
- Alternatives to PMI:
  - Positive PMI (PPMI): as PMI but 0 if PMI < 0.
  - Derivatives such as Mitchell and Lapata's (2010) weighting function (PMI without the log).

What semantic space?

- Entire vocabulary.
  - + All information included – even rare contexts
  - - Inefficient (100,000s dimensions). Noisy (e.g. *002.png|thumb|right|200px|graph_n*)
- Top *n* words with highest frequencies.
  - + More efficient (2000-10000 dimensions). Only ‘real’ words included.
  - - May miss out on infrequent but relevant contexts.

What semantic space?

- Singular Value Decomposition (LSA – Landauer and Dumais, 1997): the number of dimensions is reduced by exploiting redundancies in the data.
    - + Very efficient (200-500 dimensions). Captures generalisations in the data.
    - - SVD matrices are not interpretable.
- Other variants . . .

Outline.

Models

Getting distributions from text

Real distributions

Similarity

Distributions and classic lexical semantic relationships

Our reference text

Douglas Adams, *Mostly harmless*

The major difference between a thing that might go wrong and a thing that cannot possibly go wrong is that when a thing that cannot possibly go wrong goes wrong it usually turns out to be impossible to get at or repair.

- **Example:** Produce distributions using a word window, frequency-based model

The semantic space

Douglas Adams, *Mostly harmless*

The major difference between a thing that might go wrong and a thing that cannot possibly go wrong is that when a thing that cannot possibly go wrong goes wrong it usually turns out to be impossible to get at or repair.

- Assume only keep open-class words.
- **Dimensions:**

difference
get
go
goes
impossible
major
possibly
repair
thing
turns
usually
wrong

Frequency counts...

Douglas Adams, *Mostly harmless*

The major difference between a thing that might go wrong and a thing that cannot possibly go wrong is that when a thing that cannot possibly go wrong goes wrong it usually turns out to be impossible to get at or repair.

- **Counts:**

difference 1
get 1
go 3
goes 1
impossible 1
major 1
possibly 2
repair 1
thing 3
turns 1
usually 1
wrong 4

Conversion into 5-word windows...

Douglas Adams, *Mostly harmless*

The major difference between a thing that might go wrong and a thing that cannot possibly go wrong is that when a thing that cannot possibly go wrong goes wrong it usually turns out to be impossible to get at or repair.

- $\emptyset$ $\emptyset$ **the** major difference
- $\emptyset$ the **major** difference between
- the major **difference** between a
- major difference **between** a thing
- ...

Distribution for *wrong*

Douglas Adams, *Mostly harmless*

The major difference between a thing that [might go wrong and a] thing that cannot [possibly go wrong is that] when a thing that cannot [possibly go [wrong goes wrong] it usually] turns out to be impossible to get at or repair.

- **Distribution (frequencies):**

difference 0
get 0
go 3
goes 2
impossible 0
major 0
possibly 2
repair 0
thing 0
turns 0
usually 1
wrong 2

Distribution for *wrong*

Douglas Adams, *Mostly harmless*

The major difference between a thing that [might go wrong and a] thing that cannot [possibly go wrong is that] when a thing that cannot [possibly go [wrong goes wrong] it usually] turns out to be impossible to get at or repair.

- **Distribution (PPMIs):**

difference	0
get	0
go	0.70
goes	1
impossible	0
major	0
possibly	0.70
repair	0
thing	0
turns	0
usually	0.70
wrong	0.40

Outline.

Models

Getting distributions from text

Real distributions

Similarity

Distributions and classic lexical semantic relationships

Experimental corpus

- Dump of entire English Wikipedia (WikiWoods, 2008), parsed with the English Resource Grammar giving semantic dependencies.
- Dependencies include:
  - For nouns: head verbs (+ any other argument of the verb), modifying adjectives, head prepositions (+ any other argument of the preposition).
    e.g. cat: chase_v+mouse_n, black_a, of_p+neighbour_n
  - For verbs: arguments (NPs and PPs), adverbial modifiers.
    e.g. eat: cat_n+mouse_n, in_p+kitchen_n, fast_a
  - For adjectives: modified nouns; rest as for nouns (assuming intersective composition).
    e.g. black: cat_n, chase_v+mouse_n

System description

- Semantic space: top 100,000 contexts.
- Weighting: normalised PMI (Bouma 2007).

$$pmi_{wc} = \frac{log(\frac{f_{wc}*f_{total}}{f_w*f_c})}{-log(\frac{f_{wc}}{f_{total}})} \tag{1}$$

An example noun

- *language*:

0.54::other+than_p()+English_n
0.53::English_n+as_p()
0.52::English_n+be_v
0.49::english_a
0.48::and_c+literature_n
0.48::people_n+speak_v
0.47::French_n+be_v
0.46::Spanish_n+be_v
0.46::and_c+dialects_n
0.45::grammar_n+of_p()
0.45::foreign_a
0.45::germanic_a
0.44::German_n+be_v
0.44::of_p()+instruction_n

0.44::speaker_n+of_p()
0.42::generic_entity_rel_+speak_v
0.42::pron_rel_+speak_v
0.42::colon_v+English_n
0.42::be_v+English_n
0.42::language_n+be_v
0.42::and_c+culture_n
0.41::arabic_a
0.41::dialects_n+of_p()
0.40::part_of_rel_+speak_v
0.40::percent_n+speak_v
0.39::spanish_a
0.39::welsh_a
0.39::tonal_a

An example adjective

- *academic*:

```
0.52::Decathlon_n
0.51::excellence_n
0.45::dishonesty_n
0.45::rigor_n
0.43::achievement_n
0.42::discipline_n
0.40::vice_president_n+for_p()
0.39::institution_n
0.39::credentials_n
0.38::journal_n
0.37::journal_n+be_v
0.37::vocational_a
0.37::student_n+achieve_v
0.36::athletic_a
0.36::reputation_n+for_p()
0.35::regalia_n
0.35::program_n
0.35::freedom_n
0.35::student_n+with_p()
0.35::curriculum_n
0.34::standard_n
0.34::at_p()+institution_n
0.34::career_n
0.34::Career_n
0.33::dress_n
0.33::scholarship_n
0.33::prepare_v+student_n
0.33::qualification_n
```

Corpus choice

- As much data as possible?
  - British National Corpus (BNC): 100 m words
  - Wikipedia: 897 m words (in WikiWoods)
  - UKWac: 2 bn words
  - ...
- In general preferable, *but*:
  - More data is not necessarily the data you want.
  - More data is not necessarily realistic from a psycholinguistic point of view. We perhaps encounter 50,000 words a day. BNC = 5 years' text exposure.

Corpus choice

- Distribution for *unicycle*, as obtained from Wikipedia.

0.45::motorized_a
0.40::pron_rel_+ride_v
0.24::for_p()+entertainment_n
0.24::half_n+be_v
0.24::unwieldy_a
0.23::earn_v+point_n
0.22::pron_rel_+crash_v
0.19::man_n+on_p()
0.19::on_p()+stage_n
0.19::position_n+on_p()
0.17::slip_v
0.16::and_c+1_n
0.16::autonomous_a
0.16::balance_v
0.13::tall_a
0.12::fast_a
0.11::red_a
0.07::come_v
0.06::high_a

Polysemy

- Distribution for *pot*, as obtained from Wikipedia.

```
0.57::melt_v
0.44::pron_rel_+smoke_v
0.43::of_p()+gold_n
0.41::porous_a
0.40::of_p()+tea_n
0.39::player_n+win_v
0.39::money_n+in_p()
0.38::of_p()+coffee_n
0.33::amount_n+in_p()
0.33::ceramic_a
0.33::hot_a
0.32::boil_v
0.31::bowl_n+and_c
0.31::ingredient_n+in_p()
0.30::plant_n+in_p()
0.30::simmer_v
0.29::pot_n+and_c
0.28::bottom_n+of_p()
0.28::of_p()+flower_n
0.28::of_p()+water_n
0.28::food_n+in_p()
```

Polysemy

- Some researchers incorporate word sense disambiguation techniques.
- But most assume a single space for each word: can perhaps think of subspaces corresponding to senses.
- Graded rather than absolute notion of polysemy.

Multiword expressions

- Distribution for *time*, as obtained from Wikipedia.

```
0.46::of_p()+death_n
0.45::same_a
0.45::1_n+at_p(temp)
0.45::Nick_n+of_p()
0.42::spare_a
0.42::playoffs_n+for_p()
0.42::of_p()+retirement_n
0.41::of_p()+release_n
0.40::pron_rel_+spend_v
0.39::sand_n+of_p()
0.39::pron_rel_+waste_v
0.38::place_n+around_p()
0.38::of_p()+arrival_n
0.38::of_p()+completion_n
0.37::after_p()+time_n
0.37::of_p()+arrest_n
0.37::country_n+at_p()
0.37::age_n+at_p()
0.37::space_n+and_c
0.37::in_p()+career_n
0.37::world_n+at_p()
```

Outline.

Models

Getting distributions from text

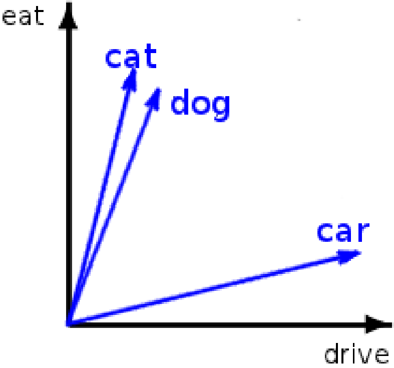
Real distributions

Similarity

Distributions and classic lexical semantic relationships

Calculating similarity in a distributional space

- Distributions are vectors, so distance can be calculated.



Measuring similarity

- Cosine:

$$\frac{\sum v1_k * v2_k}{\sqrt{\sum v1_k^2} * \sqrt{\sum v2_k^2}} \tag{2}$$

- The measure calculates the cosine of the angle between two vectors and is therefore length-independent. This is important, as frequent words have longer vectors than less frequent ones.
- Other measures include Jaccard, Lin . . .

The scale of similarity: some examples

house – building 0.43
gem – jewel 0.31
capitalism – communism 0.29
motorcycle – bike 0.29
test – exam 0.27
school – student 0.25
singer – academic 0.17
horse – farm 0.13
man –accident 0.09
tree – auction 0.02
cat –county 0.007

Words most similar to *cat*

as chosen from the 5000 most frequent nouns in Wikipedia.

1	cat
0.45	dog
0.36	animal
0.34	rat
0.33	rabbit
0.33	pig
0.31	monkey
0.31	bird
0.30	horse
0.29	mouse
0.29	wolf
0.29	creature
0.29	human
0.29	goat
0.28	snake
0.28	bear
0.28	man
0.28	cow
0.26	fox
0.26	girl
0.26	sheep
0.26	boy
0.26	elephant
0.25	deer
0.25	woman
0.25	fish
0.24	squirrel
0.24	dragon
0.24	frog
0.23	baby
0.23	child
0.23	lion
0.23	person
0.23	pet
0.23	lizard
0.23	chicken
0.22	monster
0.22	people
0.22	tiger
0.22	mammal
0.21	bat
0.21	duck
0.21	cattle
0.21	dinosaur
0.21	character
0.21	kid
0.21	turtle
0.20	robot

But what is similarity?

- In distributional semantics, very broad notion: synonyms, near-synonyms, hyponyms, taxonomical siblings, antonyms, etc.
- Correlates with a psychological reality.
- Test via correlation with human judgments on the Miller & Charles (1991) test set.
- M&C was re-run of Rubenstein & Goodenough (1965). Correlation coefficient between M&C and R&G = 0.97.

Miller & Charles 1991

3.92	automobile-car
3.84	journey-voyage
3.84	gem-jewel
3.76	boy-lad
3.7	coast-shore
3.61	asylum-madhouse
3.5	magician-wizard
3.42	midday-noon
3.11	furnace-stove
3.08	food-fruit
3.05	bird-cock
2.97	bird-crane
2.95	implement-tool
2.82	brother-monk
1.68	crane-implement
1.66	brother-lad
1.16	car-journey
1.1	monk-oracle
0.89	food-rooster
0.87	coast-hill
0.84	forest-graveyard
0.55	monk-slave
0.42	lad-wizard
0.42	coast-forest
0.13	cord-smile
0.11	glass-magician
0.08	rooster-voyage
0.08	noon-string

- Distributional systems, reported correlations 0.8 or more.

TOEFL synonym test

Test of English as a Foreign Language: task is to find the best match to a word:

Prompt:		levied
Choices:	(a)	imposed
	(b)	believed
	(c)	requested
	(d)	correlated
Solution:	(a)	imposed

- Non-native English speakers applying to college in US reported to average 65%
- Best corpus-based results are 100%

Similarity: some things to watch out for

- Correlation with similarity datasets is the standard way of evaluating a distributional model.
- Very few papers discuss significance testing: this is not straightforward to do properly.
- Huge parameter space for distributional models: exhaustive search is problematic (xkcd jelly beans xkcd.com/882/).
- Similarity versus relatedness, e.g., SimLex-999 (but inconsistencies).
- Human similarity is best case: 2.97 bird-crane (presumably also high similarity for crane-hoist but not bird-hoist). This disadvantages some potentially useful models.

Similarity: more things to watch

- Split a corpus into two (randomly), calculate distribution for word X on each half, check similarity between distibutions of X: what happens?
- Use unlemmatized corpus (as in most experiments), calculate similarity between inflectional forms of words (e.g., *cat* vs *cats*, *sleep* vs *sleeping*): what happens?

Outline.

Models

Getting distributions from text

Real distributions

Similarity

Distributions and classic lexical semantic relationships

Distributional methods are a usage representation

- Distributions are a good conceptual representation if you believe that ‘the meaning of a word is given by its usage’.
- Corpus-dependent, culture-dependent, register-dependent.
  Example: calculated similarity between *policeman* and *cop*: 0.23

Distribution for *policeman*

policeman
0.59::ball_n+poss_rel
0.48::and_c+civilian_n
0.42::soldier_n+and_c
0.41::and_c+soldier_n
0.38::secret_a
0.37::people_n+include_v
0.37::corrupt_a
0.36::uniformed_a
0.35::uniform_n+poss_rel
0.35::civilian_n+and_c
0.31::iraqi_a
0.31::lot_n+poss_rel
0.31::chechen_a
0.30::laugh_v
0.29::and_c+criminal_n
0.28::incompetent_a
0.28::pron_rel_+shoot_v
0.28::hat_n+poss_rel
0.28::terrorist_n+and_c
0.27::and_c+crowd_n
0.27::military_a
0.27::helmet_n+poss_rel
0.27::father_n+be_v
0.26::on_p()+duty_n
0.25::salary_n+poss_rel
0.25::on_p()+horseback_n
0.25::armed_a
0.24::and_c+nurse_n
0.24::job_n+as_p()
0.24::open_v+fire_n

Distribution for *cop*

cop

0.45::crooked_a
0.45::corrupt_a
0.44::maniac_a
0.38::dirty_a
0.37::honest_a
0.36::uniformed_a
0.35::tough_a
0.33::pron_rel_+call_v
0.32::funky_a
0.32::bad_a
0.29::veteran_a
0.29::and_c+robot_n
0.28::and_c+criminal_n
0.28::bogus_a
0.28::talk_v+to_p()+pron_rel_
0.27::investigate_v+murder_n
0.26::on_p()+force_n
0.25::parody_n+of_p()
0.25::Mason_n+and_c
0.25::pron_rel_+kill_v
0.25::racist_a
0.24::addicted_a
0.23::gritty_a
0.23::and_c+interference_n
0.23::arrive_v
0.23::and_c+detective_n
0.22::look_v+way_n
0.22::dead_a
0.22::pron_rel_+stab_v
0.21::pron_rel_+evade_v

The similarity of synonyms

- Similarity between *egglant/aubergine* from Wikipedia: 0.11 Relatively low cosine. Partly due to frequency (222 for *eggplant*, 56 for *aubergine*).
- Similarity between *policeman/cop*: 0.23
- Similarity between *city/town*: 0.73

In general, true synonymy does not correspond to higher similarity scores than near-synonymy.

Similarity of antonyms

- Similarities between:
  - cold/hot 0.29
  - dead/alive 0.24
  - large/small 0.68
  - colonel/general 0.33

Identifying antonyms

- Antonyms have high distributional similarity: hard to distinguish from near-synonyms purely by distributions.
- Identification by heuristics applied to pairs of highly similar distributions.
- For instance, antonyms are frequently coordinated while synonyms are not:
    - a selection of cold and hot drinks
    - wanted dead or alive
    - lectures, readers and professors are invited to attend

Lexical semantics: slide from lecture 7

- Limited domain: mapping to some knowledge base term(s). Knowledge base constrains possible meanings.
- Issues for broad coverage systems:
  - Boundary between lexical meaning and world knowledge.
  - Representing lexical meaning.
  - Acquiring representations.
  - Polysemy and multiword expressions.

Distributional semantics: some conclusions

- Boundary between lexical meaning and world knowledge.
  Ignored: use whatever turns up in the distribution.
- Representing lexical meaning.
  Vector (more generally tensor).
- Acquiring representations.
  Extract from corpora.
- Polysemy and multiword expressions.
  Multiple senses in single distribution, MWEs in distribution.
- Also: usually much more syntax-sensitive than classical lexical semantics.

Distributions are partial lexical semantic representations, but very useful and theoretically interesting.

Next lecture

- embeddings in neural architectures
- word2vec (see also Paula Buttery's notes) and doc2vec
- some visualization techniques
- if there's time: a bit about multimodal systems and visual question answering