# Word Meaning: <br> Distributional Representations \& Word Sense Disambiguation 

CMSC 723 / LING 723 / INST 725

Marine Carpuat

## Reminders

- Read the syllabus
- Make sure you have access to piazza
- Get started on homework 1 - due Thursday Sep 7 by 12 pm.


## Today: Word Meaning

2 core issues from an NLP perspective

- Semantic similarity: given two words, how similar are they in meaning?
- Word sense disambiguation: given a word that has more than one meaning, which one is used in a specific context?


# Word similarity for question answering 

"fast" is similar to "rapid"
"tall" is similar to "height"

Question answering:
Q: "How tall is Mt. Everest?"
Candidate A: "The official height of Mount Everest is 29029 feet"

# Word similarity for plagiarism detection 

## MAINFRAMES

Mainframes are primarily referred to large computers with rapid, advanced processing capabilities that can execute and perform tasks equivalent to many Personal Computers (PCs) machines networked together. It is characterized with high quantity Random Access Memory (RAM), very large secondary storage devices, and high-speed processors to cater for the needs of the computers under its service.
Consisting of advanced components, mainframes have the capability of running multiple large applications required by many and most enterprises and organizations. This is one of its advantages. Mainframes are also suitable to cater for those applications (programs) or files that are of very high demand by its users (clients).
Examples of such organizations and enterprises using mainframes are online shopping websites such as Fhav $\Delta$ masen and nomnıtinn_diant

## MAINFRAMES

Mainframes usually are referred those computers with fast, advanced processing capabilities that could perform by itself tasks that may require a lot of Personal Computers (PC) Machines. Usually mainframes would have lots of RAMs, very large secondary storage devices, and very fast processors to cater for the needs of those computers under its service.
Due to the advanced components mainframes have, these computers have the capability of running multiple large applications required by most enterprises, which is one of its advantage. Mainframes are also suitable to cater for those applications or files that are of very large demand by its users (clients). Examples of these include the large online shopping websites -i.e. : Ebay, Amazon, Microsoft, etc.

Word similarity for historical linguistics: semantic change over time


Kulkarni, Al-Rfou, Perozzi, Skiena 2015

## tesguiino

A bottle of tesguiino is on the table Everybody likes tesguiino Tesgüino makes you drunk We make tesguiino out of corn.

Intuition: two words are similar if they have similar word contexts.

Embedding word meaning in vector space
Vector Semantics

## Distributional models of meaning <br> = vector-space models of meaning <br> = vector semantics

## Intuitions

Zellig Harris (1954):

- "oculist and eye-doctor ... occur in almost the same environments"
- "If A and B have almost identical environments we say that they are synonyms."

Firth (1957):

- "You shall know a word by the company it keeps!"


## Vector Semantics

- Model the meaning of a word by "embedding" in a vector space.
- The meaning of a word is a vector of numbers
- Vector models are also called "embeddings".
- Contrast: word meaning is represented in many NLP applications by a vocabulary index ("word number 545")


## Many varieties of vector models

Sparse vector representations

1. Mutual-information weighted word co-occurrence matrices

Dense vector representations:
2. Singular value decomposition (and Latent Semantic Analysis)
3. Neural-network-inspired models (skip-grams, CBOW)

## Term-document matrix

- Each cell: count of term $t$ in a document $d$ : $\mathrm{tf}_{t, d}$
- Each document is a count vector in $\mathbb{N}^{v}$ : a column below

|  | As You Like It | Twelfth Night | Julius Caesar | Henry V |
| :--- | ---: | ---: | ---: | ---: |
| battle | 1 | 1 | 8 | 15 |
| soldier | 2 | 2 | 12 | 36 |
| fool | 37 | 58 | 1 | 5 |
| clown | 6 | 117 | 0 | 0 |

## Term-document matrix

- Two documents are similar if their vectors are similar

|  | As You Like It | Twelfth Night | Julius Caesar | Henry V |
| :--- | ---: | ---: | ---: | ---: |
| battle | 1 | 1 | 8 <br> 12 <br> soldier | 2 |

## The words in a term-document matrix

- Each word is a count vector in $\mathbb{N}^{D}$ : a row below

|  | As You Like It | Twelfth Night | Julius Caesar | Henry V |
| :--- | ---: | ---: | ---: | ---: |
| battle | 1 | 1 | 8 | 15 |
| soldier | 2 | 2 | 12 | 36 |
| fool | 37 | 58 | 1 | 5 |
| clown | 6 | 117 | 0 | 0 |

## The words in a term-document matrix

- Two words are similar if their vectors are similar

|  | As You Like It | Twelfth Night | Julius Caesar | Henry V |
| :--- | ---: | ---: | ---: | ---: |
| battle | 1 | 1 | 8 | 15 |
| soldier | 2 | 2 | 12 | 36 |
| fool | 37 | 58 | 1 | 5 |
| clown | 6 | 117 | 0 | 0 |

## The word-word or word-context matrix

- Instead of entire documents, use smaller contexts
- Paragraph
- Window of $\pm 4$ words
- A word is now defined by a vector over counts of context words
- Instead of each vector being of length D
- Each vector is now of length |V|
- The word-word matrix is $|\mathrm{V}| \mathrm{x}|\mathrm{V}|$


# Word-Word matrix <br> Sample contexts $\pm 7$ words 


#### Abstract

sugar, a sliced lemon, a tablespoonful of apricot their enjoyment. Cautiously she sampled her first pineapple well suited to programming on the digital computer. preserve or jam, a pinch each of, and another fruit whose taste she likened In finding the optimal R-stage policy from for the purpose of gathering data and information necessary for the study authorized in the


|  | aardvark | computer | data | pinch | result | sugar | $\ldots$ |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 0 | 0 | 0 | 1 | 0 | 1 |  |
| apricot | 0 | 0 | 0 | 1 | 0 | 1 |  |
| pineapple | 0 | 2 | 1 | 0 | 1 | 0 |  |
| digital | 0 | 1 | 6 | 0 | 4 | 0 |  |

## Word-word matrix

- The $|\mathrm{V}| \mathrm{x}|\mathrm{V}|$ matrix is very sparse (most values are 0 )
- The size of windows depends on representation goals
- The shorter the windows, the more syntactic the representation $\pm 1-3$ very syntacticy
- The longer the windows, the more semantic the representation $\pm 4-10$ more semanticy


# Positive Pointwise Mutual Information (PPMI) 

Vector Semantics

## Problem with raw counts

- Raw word frequency is not a great measure of association between words
- We'd rather have a measure that asks whether a context word is particularly informative about the target word.
- Positive Pointwise Mutual Information (PPMI)


## Pointwise Mutual Information

## Pointwise mutual information:

Do events $x$ and $y$ co-occur more than if they were independent?

$$
\operatorname{PMI}(X, Y)=\log _{2} \frac{P(x, y)}{P(x) P(y)}
$$

## PMI between two words: (Church \& Hanks 1989)

Do words $x$ and $y$ co-occur more than if they were independent?

$$
\operatorname{PMI}\left(\text { word }_{1}, \text { word }_{2}\right)=\log _{2} \frac{P\left(\text { word }_{1}, \text { word }_{2}\right)}{P\left(\text { word }_{1}\right) P\left(\text { word }_{2}\right)}
$$

## Positive Pointwise Mutual Information

- PMI ranges from $-\infty$ to $+\infty$
- But the negative values are problematic
- Things are co-occurring less than we expect by chance
- Unreliable without enormous corpora
- So we just replace negative PMI values by 0
- Positive PMI (PPMI) between word1 and word2:

$$
\operatorname{PPMI}\left(\text { word }_{1}, \text { word }_{2}\right)=\max \left(\log _{2} \frac{P\left(\text { word }_{1}, \text { word }_{2}\right)}{P\left(\text { word }_{1}\right) P\left(\text { word }_{2}\right)}, 0\right)
$$

## Computing PPMI on a term-context matrix

- Matrix F with W rows (words) and C columns (contexts)
- $\mathrm{f}_{\mathrm{ij}}$ is \# of times $\mathrm{w}_{\mathrm{i}}$ occurs in context $\mathrm{c}_{\mathrm{j}}$
apricot
pineapple
digital
information

| aardvark | computer | data | pinch | result | sugar |
| :---: | :---: | :---: | :---: | :---: | :---: |
| 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 2 | 1 | 0 | 1 | 0 |
| 0 | 1 | 6 | 0 | 4 | 0 |

$$
\begin{gathered}
p_{i j}=\frac{f_{i j}}{\sum_{i=1}^{W} \sum_{j=1}^{C} f_{i j}} \quad p_{i^{*}}=\frac{\sum_{j=1}^{C} f_{i j}}{\sum_{i=1}^{W} \sum_{j=1}^{C} f_{i j}} \quad p_{* j}=\frac{\sum_{i=1}^{W} f_{i j}}{\sum_{i=1}^{W} \sum_{j=1}^{C} f_{i j}} \\
p m i_{i j}=\log _{2} \frac{p_{i j}}{p_{i^{*}} p_{*_{j}}} \quad \text { ppmi } i_{i j}=\left\{\begin{array}{cc}
p m i_{i j} & \text { if } p m i_{i j}>0 \\
0 & \text { otherwise }
\end{array}\right.
\end{gathered}
$$

## Count(w,context)

$$
p_{i j}=\begin{array}{llrrrrr}
f_{i j} & & \text { computer } & \text { data } & \text { pinch } & \text { result } & \text { sugar } \\
\sum_{i=1}^{W} \sum_{j=1}^{C} f_{i j} & \text { apricot } & 0 & 0 & 1 & 0 & 1 \\
& \text { pineapple } & 0 & 0 & 1 & 0 & 1 \\
& \text { digital } & 2 & 1 & 0 & 1 & 0 \\
& \text { information } & 1 & 6 & 0 & 4 & 0
\end{array}
$$

$p(w=$ information,$c=$ data $)=6 / 19=.32$
$p(w=$ information $)=11 / 19=.58$
$p(c=$ data $)=7 / 19=.37$
$p\left(w_{i}\right)=\frac{\sum_{j=1}^{C} f_{i j}}{N} \quad p\left(c_{j}\right)=\frac{\sum_{i=1}^{W} f_{i j}}{N}$

|  | p(w,context) |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | computer | data | pinch | result | sugar | p(w) |
| apricot | 0.00 | 0.00 | 0.05 | 0.00 | 0.05 | 0.11 |
| pineapple | 0.00 | 0.00 | 0.05 | 0.00 | 0.05 | 0.11 |
| digital | 0.11 | 0.05 | 0.00 | 0.05 | 0.00 | 0.21 |
| information | 0.05 | 0.32 | 0.00 | 0.21 | 0.00 | 0.58 |
|  |  |  |  |  |  |  |
| p(context) | 0.16 | 0.37 | 0.11 | 0.26 | 0.11 |  |



## Weighting PMI

- PMI is biased toward infrequent events
- Very rare words have very high PMI values
- Two solutions:
- Give rare words slightly higher probabilities
- Use add-k smoothing (which has a similar effect)


## Weighting PMI: Giving rare context words slightly higher probability

- Raise the context probabilities to $\alpha=0.75$ :

$$
\begin{aligned}
\operatorname{PPMI}_{\alpha}(w, c) & =\max \left(\log _{2} \frac{P(w, c)}{P(w) P_{\alpha}(c)}, 0\right) \\
P_{\alpha}(c) & =\frac{\operatorname{count}(c)^{\alpha}}{\sum_{c} \operatorname{count}(c)^{\alpha}}
\end{aligned}
$$

- Consider two events, $\mathrm{P}(\mathrm{a})=.99$ and $\mathrm{P}(\mathrm{b})=.01$

$$
P_{\alpha}(a)=\frac{.99 .75}{.99 .75+.01^{.75}}=.97 P_{\alpha}(b)=\frac{.01^{.75}}{.01^{.75}+.01^{.75}}=.03
$$

## Add-2 smoothing

## Add-2 Smoothed Count(w,context

|  | computer | data | pinch | result | sugar |
| :--- | ---: | ---: | ---: | ---: | ---: |
| apricot | 2 | 2 | 3 | 2 | 3 |
| pineapple | 2 | 2 | 3 | 2 | 3 |
| digital | 4 | 3 | 2 | 3 | 2 |
| information | 3 | 8 | 2 | 6 | 2 |

## PPMI vs add-2 smoothed PPMI

|  | PPMI(w,context) |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | computer | data | pinch | result | sugar |
| apricot | - |  | 2.25 |  | 2.25 |
| pineapple | - | - | 2.25 |  | 2.25 |
| digital | 1.66 | 0.00 |  | 0.00 | - |
| information | 0.00 | 0.57 |  | 0.47 | - |
|  | PPMI(w,context) [add-2] |  |  |  |  |
|  | computer | data | pinch | result | sugar |
| apricot | 0.00 | 0.00 | 0.56 | 0.00 | 0.56 |
| pineapple | 0.00 | 0.00 | 0.56 | 0.00 | 0.56 |
| digital | 0.62 | 0.00 | 0.00 | 0.00 | 0.00 |
| information | 0.00 | 0.58 | 0.00 | 0.37 | 0.00 |

## tf.idf: an alternative to PPMI for measuring association

- The combination of two factors
- TF: Term frequency (Luhn 1957): frequency of the word
- IDF: Inverse document frequency (Sparck Jones 1972)
- N is the total number of documents
- $\mathrm{df}_{\mathrm{i}}=$ "document frequency of word $i$ "
= \# of documents with word $i$

$$
\operatorname{idf}_{i}=\log \left(\frac{N}{d f_{i}}\right)
$$

- $w_{i j}=$ word $i$ in document $j$

$$
w_{i j}=t f_{i j} i d f_{i}
$$

Measuring similarity: the cosine
Vector Semantics

## Cosine for computing similarity


$v_{i}$ is the PPMI value for word $v$ in context $i$
$w_{i}$ is the PPMI value for word $w$ in context $i$.
$\operatorname{Cos}(\overrightarrow{v, w})$ is the cosine similarity of $\vec{v}$ and $\vec{w}$

## Other possible similarity measures

$$
\begin{aligned}
& \operatorname{sim}_{\text {Dice }}(\vec{v}, \vec{w})=\frac{2 \times \sum_{i=1}^{N} \min \left(v_{i}, w_{i}\right)}{\sum_{i=1}^{N}\left(v_{i}+w_{i}\right)} \\
& \operatorname{sim}_{\mathrm{JS}}(\vec{v}| | \vec{w}) \quad=D\left(\vec{v} \left\lvert\, \frac{i=1}{2}+\vec{w}\right.\right)+D\left(\vec{w} \left\lvert\, \frac{\vec{v}+\vec{w}}{2}\right.\right)
\end{aligned}
$$

## Evaluating similarity

Vector Semantics

## Evaluating similarity

- Extrinsic (task-based, end-to-end) Evaluation:
- Question Answering
- Spell Checking
- Essay grading
- Intrinsic Evaluation:
- Correlation between algorithm and human word similarity ratings
- Wordsim353: 353 noun pairs rated 0-10. sim(plane,car)=5.77
- Taking TOEFL multiple-choice vocabulary tests
- Levied is closest in meaning to:
imposed, believed, requested, correlated


## Today: Word Meaning

2 core issues from an NLP perspective

- Semantic similarity: given two words, how similar are they in meaning?
- Word sense disambiguation: given a word that has more than one meaning, which one is used in a specific context?


## "Big rig carrying fruit crashes on 210 Freeway, creates jam"

http://articles.latimes.com/2013/may/20/local/la-me-In-big-rig-crash-20130520

# How do we know that a word (lemma) has distinct senses? 

- Linguists often design tests for this purpose
- e.g., zeugma combines distinct senses in an uncomfortable way

Which flight serves breakfast?

Which flights serve BWI?
*Which flights serve breakfast and BWI?

## Word Senses

- "Word sense" = distinct meaning of a word
- Same word, different senses
- Homonyms (homonymy): unrelated senses; identical orthographic form is coincidental
- E.g., financial bank vs. river bank
- Polysemes (polysemy): related, but distinct senses
- E.g., Financial bank vs. blood bank vs. tree bank
- Metonyms (metonymy): "stand in", technically, a sub-case of polysemy
- E.g., use "Washington" in place of "the US government"
- Different word, same sense
- Synonyms (synonymy)
- Homophones: same pronunciation, different orthography, different meaning
- Examples: would/wood, to/too/two
- Homographs: distinct senses, same orthographic form, different pronunciation
- Examples: bass (fish) vs. bass (instrument)


## Relationship Between Senses

- IS-A relationships
- From specific to general (up): hypernym (hypernymy)
- From general to specific (down): hyponym (hyponymy)
- Part-Whole relationships
- wheel is a meronym of car (meronymy)
- car is a holonym of wheel (holonymy)


## WordNet: a lexical database for English

## https://wordnet.princeton.edu/

- Includes most English nouns, verbs, adjectives, adverbs
- Electronic format makes it amenable to automatic manipulation: used in many NLP applications
- "WordNets" generically refers to similar resources in other languages


## Synonymy in WordNet

- WordNet is organized in terms of "synsets"
- Unordered set of (roughly) synonymous "words" (or multi-word phrases)
- Each synset expresses a distinct meaning/concept


## WordNet: Example

Noun
\{pipe, tobacco pipe\} (a tube with a small bowl at one end; used for smoking tobacco)
\{pipe, pipage, piping\} (a long tube made of metal or plastic that is used to carry water or oil or gas etc.)
\{pipe, tube\} (a hollow cylindrical shape)
\{pipe\} (a tubular wind instrument)
\{organ pipe, pipe, pipework\} (the flues and stops on a pipe organ)

Verb
\{shriek, shrill, pipe up, pipe\} (utter a shrill cry)
\{pipe\} (transport by pipeline) "pipe oil, water, and gas into the desert"
\{pipe\} (play on a pipe) "pipe a tune"
\{pipe\} (trim with piping) "pipe the skirt"

## WordNet 3.0: Size

| Part of speech | Word form | Synsets |
| ---: | ---: | ---: |
| Noun | 117,798 | 82,115 |
| Verb | 11,529 | 13,767 |
| Adjective | 21,479 | 18,156 |
| Adverb | 4,481 | 3,621 |
| Total | 155,287 | 117,659 |

## Word Sense Disambiguation

- Task: automatically select the correct sense of a word
- Input: a word in context
- Output: sense of the word
- Motivated by many applications:
- Information retrieval
- Machine translation
- ...


## How big is the problem?

- Most words in English have only one sense
- 62\% in Longman's Dictionary of Contemporary English
- 79\% in WordNet
- But the others tend to have several senses
- Average of 3.83 in LDOCE
- Average of 2.96 in WordNet
- Ambiguous words are more frequently used
- In the British National Corpus, $84 \%$ of instances have more than one sense
- Some senses are more frequent than others


## Baseline Performance

## - Baseline: most frequent sense

- Equivalent to "take first sense" in WordNet
- Does surprisingly well!

| Freq Synset | Gloss |
| :---: | :---: |
| 338 plant $^{1}$, works, industrial plant | buildings for carrying on industrial labor |
| ( 207 plant ${ }^{2}$, flora, plant life | a living organism lacking the power of locomotion |
| - 2 plant ${ }^{3}$ | something planted secretly for discovery by another |
| $\therefore$ plant ${ }^{4}$ | an actor situated in the audience whose acting is rehearsed but seems spontaneous to the audience |

$62 \%$ accuracy in this case!

## Upper Bound Performance

- Upper bound
- Fine-grained WordNet sense: $75-80 \%$ human agreement
- Coarser-grained inventories: 90\% human agreement possible


## Simplest WSD algorithm: Lesk's Algorithm

- Intuition: note word overlap between context and dictionary entries
- Unsupervised, but knowledge rich
 because it invests in adjustable-rate mortgagése-curitiè:

| bank $^{1}$ | Gloss: | a financial institution that accepts (deposits <br> money into lending activities <br> "he cashed a check at the bank", "that bank holds the (mortgage, <br> on my home" |
| :--- | :--- | :--- |
| bank $^{2}$ | Gloss: <br> Examples: | sloping land (especially the slope beside a body of water) <br> "they pulled the canoe up on the bank", "he sat on the bank of <br> the river and watched the currents" |

## Lesk's Algorithm

- Simplest implementation:
- Count overlapping content words between glosses and context
- Lots of variants:
- Include the examples in dictionary definitions
- Include hypernyms and hyponyms
- Give more weight to larger overlaps (e.g., bigrams)
- Give extra weight to infrequent words
- ...


## Alternative: WSD as Supervised Classification



## Existing Corpora

- Lexical sample
- line-hard-serve corpus (4k sense-tagged examples)
- interest corpus (2,369 sense-tagged examples)
- ...
- All-words
- SemCor (234k words, subset of Brown Corpus)
- Senseval/SemEval (2081 tagged content words from 5k total words)
- ...


## Word Meaning

2 core issues from an NLP perspective

- Semantic similarity: given two words, how similar are they in meaning?
- Key concepts: vector semantics, PPMI and its variants, cosine similarity
- Word sense disambiguation: given a word that has more than one meaning, which one is used in a specific context?
- Key concepts: word sense, WordNet and sense inventories, unsupervised disambiguation (Lesk), supervised disambiguation

