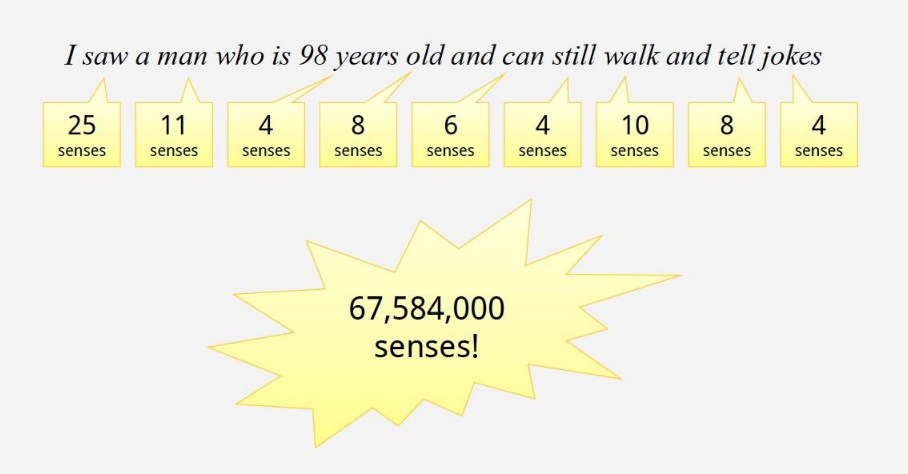
Word Sense Disambiguation

Problem in mapping words to wordnet senses

I saw a man who is 98 years old and can still walk and tell jokes

Ambiguity is rampant!



Word Sense Disambiguation (WSD)

Sense ambiguity

- Many words have several meanings or senses
- The meaning of **bass** depends on the context
- Are we talking about music, or fish?
  - An electric guitar and **bass** player stand off to one side, not really part of the scene, just as a sort of nod to gringo expectations perhaps.
  - And it all started when fishermen decided the striped **bass** in Lake Mead were too skinny.

Disambiguation

- The task of disambiguation is to determine which of the senses of an ambiguous word is invoked in a particular use of the word.
- This is done by looking at the context of the word's use.

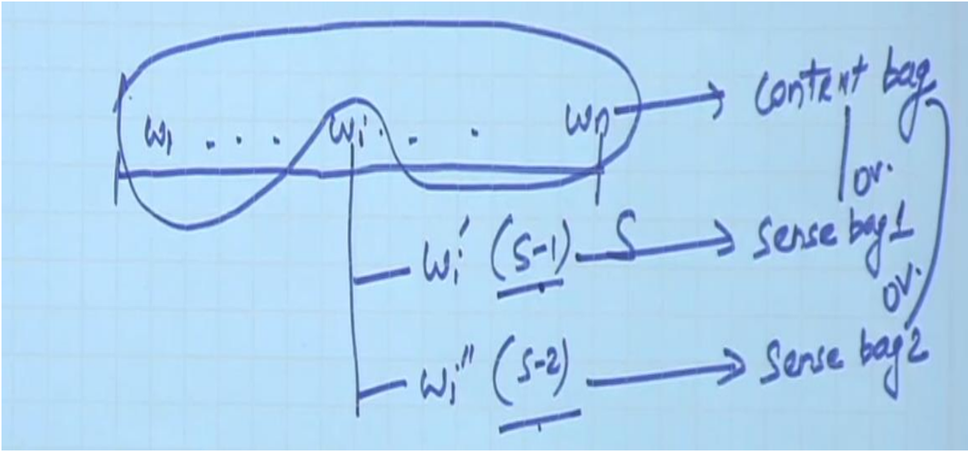
Algorithms

- Knowledge Based Approaches
  - Overlap Based Approaches
- Machine Learning Based Approaches
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms
- Hybrid Approaches

Knowledge Based Approaches

Overlap Based Approaches

- Require a **Machine Readable Dictionary** (MRD).
- Find the overlap between the features of different senses of an ambiguous word (**sense bag**) and the features of the words in its context (**context bag**).
- The features could be sense definitions, example sentences, hypernyms etc.
- The features could also be given weights.
- The sense which has the maximum overlap is selected as the contextually appropriate sense.



w1 . . . wi . . . wn
content bag
ov.
wi' (S-1)
S
sense bag 1
ov.
wi'' (S-2)
sense bag 2

Lesk's Algorithm

Sense Bag: contains the words in the definition of a candidate sense of the ambiguous word.
Context Bag: contains the words in the definition of each sense of each context word.

*On burning **coal** we get ash.*

Ash

- Sense 1
  Trees of the olive family with pinnate leaves, thin furrowed bark and gray branches.
- Sense 2
  The ***solid*** residue left when ***combustible*** material is thoroughly ***burn***ed or oxidized.
- Sense 3
  To convert into ash

Coal

- Sense 1
  A piece of glowing carbon or ***burn***t wood.
- Sense 2
  charcoal.
- Sense 3
  A black ***solid combustible*** substance formed by the partial decomposition of vegetable matter without free access to air and under the influence of moisture and often increased pressure and temperature that is widely used as a fuel for ***burn***ing

In this case Sense 2 of ash would be the winner sense.

Disambiguation based on sense definition

Lesk algorithm

A word's dictionary definitions are likely to be good indicators for the senses they define.

- Retrieve all sense definitions of target word
- Compare with sense definitions of words in context
- Choose the sense with the most overlapping words

Example

pine

a kind of **evergreen tree** with needle-shaped leaves
to waste away through sorrow or illness

cone

A solid body which narrows to a point
Something of this shape, whether solid or hollow
Fruit of certain **evergreen trees**

Lesk in action

Using the Lesk algorthm and the simplified dictionary below, disambiguate each word in the sentence: *Bonds rise when interest rates drop.*

bond
1. an electrical force linking atoms
2. a certificate of debt issued in order to raise money
3. a connection based on kinship or common interest

rise
1. move upward
2. increase in value or to a higher point
3. stand up; assume an upright position

interest
1. a sense of curiosity about someone or something
2. a reason for wanting something done
3. a fixed charge for borrowing money

drop
1. let fall to the ground
2. fall vertically
3. go down in value

rate
1. a magnitude or frequency relative to a time unit
2. amount of a charge or payment relative to some basis
3. the relative speed of progress or change

(Note: function words and inflections should be ignored.)

Walker's Algorithm

- A Thesaurus Based approach
- **Step 1:** For each sense of the target word find the thesaurus category to which that sense belongs
- **Step 2:** Calculate the score for each sense by using the context words. *A context word will add 1 to the score of the sense if the thesaurus category of the word matches that of the sense.*
  - E.g. The money in this bank fetches an interest of 8% per annum
  - Target word: *bank*
  - Clue words from the context: *money, interest, annum, fetch*

	Sense1: Finance	Sense2: Location
Money	+1	0
Interest	+1	0
Fetch	0	0
Annum	+1	0
Total	3	0

Context words add 1 to the sense when the topic of the word matches that of the sense

WSD Using Random Walk Algorithm

The **church bells** no longer **rung** on **Sundays**.

church

1: one of the groups of Christians who have their own beliefs and forms of worship
2: a place for public (especially Christian) worship
3: a service conducted in a church

bell

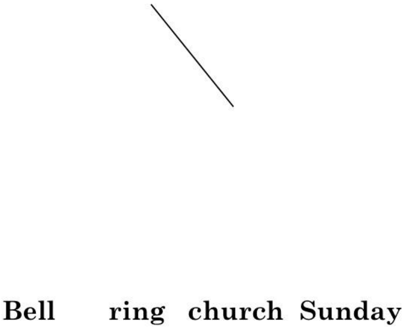
1: a hollow device made of metal that makes a ringing sound when struck
2: a push button at an outer door that gives a ringing or buzzing signal when pushed
3: the sound of a bell

ring

1: make a ringing sound
2: ring or echo with sound
3: make (bells) ring, often for the purposes of musical edification

Sunday

1: first day of the week; observed as a day of rest and worship by most Christians



WSD Using Random Walk Algorithm

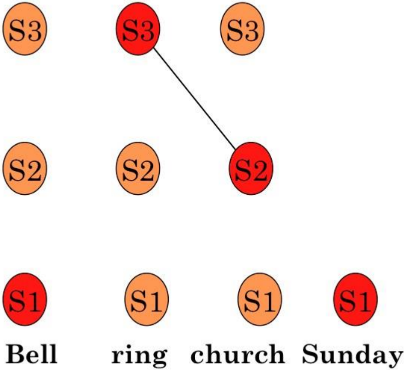
The **church bells** no longer **rung** on **Sundays**.

church
1: one of the groups of Christians who have their own beliefs and forms of worship
2: a place for public (especially Christian) worship
3: a service conducted in a church

bell
1: a hollow device made of metal that makes a ringing sound when struck
2: a push button at an outer door that gives a ringing or buzzing signal when pushed
3: the sound of a bell

ring
1: make a ringing sound
2: ring or echo with sound
3: make (bells) ring, often for the purposes of musical edification

Sunday
1: first day of the week; observed as a day of rest and worship by most Christians



Step 1: Add a vertex for each possible sense of each word in the text.

WSD Using Random Walk Algorithm

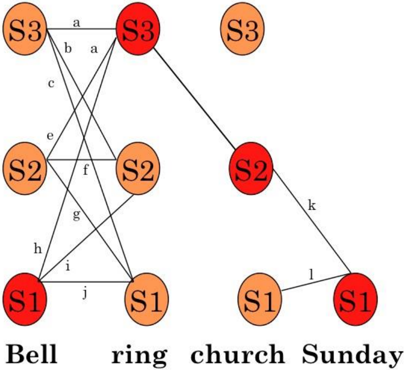
The **church bells** no longer **rung** on **Sundays**.

church
1: one of the groups of Christians who have their own beliefs and forms of worship
2: a place for public (especially Christian) worship
3: a service conducted in a church

bell
1: a hollow device made of metal that makes a ringing sound when struck
2: a push button at an outer door that gives a ringing or buzzing signal when pushed
3: the sound of a bell

ring
1: make a ringing sound
2: ring or echo with sound
3: make (bells) ring, often for the purposes of musical edification

Sunday
1: first day of the week; observed as a day of rest and worship by most Christians



Step 2: Add weighted edges using definition based semantic similarity (Lesk's method).

WSD Using Random Walk Algorithm

The **church bells** no longer **rung** on **Sundays**.

church

1: one of the groups of Christians who have their own beliefs and forms of worship
2: a place for public (especially Christian) worship
3: a service conducted in a church

bell

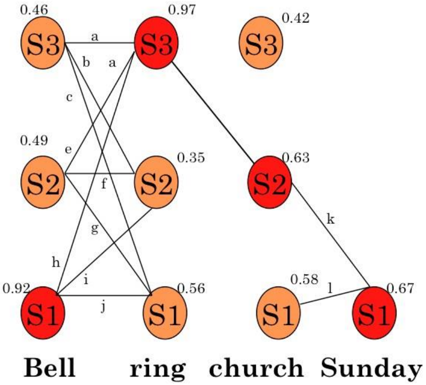
1: a hollow device made of metal that makes a ringing sound when struck
2: a push button at an outer door that gives a ringing or buzzing signal when pushed
3: the sound of a bell

ring

1: make a ringing sound
2: ring or echo with sound
3: make (bells) ring, often for the purposes of musical edification

Sunday

1: first day of the week; observed as a day of rest and worship by most Christians



Step 3: Apply graph based ranking algorithm to find score of each vertex (i.e. for each word sense).

WSD Using Random Walk Algorithm

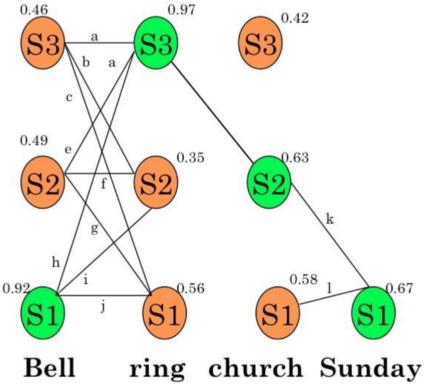
The **church bells** no longer **rung** on **Sundays**.

church
1: one of the groups of Christians who have their own beliefs and forms of worship
2: a place for public (especially Christian) worship
3: a service conducted in a church

bell
1: a hollow device made of metal that makes a ringing sound when struck
2: a push button at an outer door that gives a ringing or buzzing signal when pushed
3: the sound of a bell

ring
1: make a ringing sound
2: ring or echo with sound
3: make (bells) ring, often for the purposes of musical edification

Sunday
1: first day of the week; observed as a day of rest and worship by most Christians



Step 4: Select the vertex (sense) which has the highest score.

Naïve Bayes for WSD

- A Naïve Bayes classifer chooses the most likely sense for a word given the features of the context:

$$\hat{s} =_{s \in S} P(s|\,f)$$

- Using Bayes' law, this can be expressed as:

$$\hat{s} =_{s \in S} \frac{P(s)P(f|s)}{P(f)}$$

$$=_{s \in S} P(s)P(\,f|s)$$

- The 'Naïve' assumption: all the features are conditionally independent, given the sense':

$$\hat{s} =_{s \in S} P(s) \prod_{j=1}^{n} P(f_j|s)$$

Training for Naïve Bayes

- 'f' is a feature vector consisting of:
  - POS of w
  - Semantic and Syntactic features of w
  - Collocation vector (set of words around it) → next word (+1), +2, -1, -2 and their POS's
  - Co-occurrence vector
- Set parameters of Naïve Bayes using maximum likelihood estimation (MLE) from training data

$$P(s_i) = \frac{count(s_i, w_j)}{count(w_j)}$$

$$P(f_j | s_i) = \frac{count(f_j, s_i)}{count(s_i)}$$

Decision List Algorithm

- Based on ‘One sense per collocation’ property
  - Nearby words provide strong and consistent clues as to the sense of a target word
- Collect a large set of collocations for the ambiguous word
- Calculate word-sense probability distributions for all such collocations
- Calculate the log-likelihood ratio

$$log\left(\frac{P(Sense-A|Collocation_i)}{P(Sense-B|Collocation_i)}\right)$$

- Higher log-likelihood $\Rightarrow$ more predictive evidence
- Collocations are ordered in a decision list, with most predictive collocations ranked highest

Decision List Algorithm

Training Data

Sense	Training Examples (Keyword in Context)
A	used to strain microscopic *plant* life from the ...
A	... zonal distribution of *plant* life . ...
A	close-up studies of *plant* life and natural ...
A	too rapid growth of aquatic *plant* life in water ...
A	... the proliferation of *plant* and animal life ...
A	establishment phase of the *plant* virus life cycle ...
A	... ...
B	... ...
B	computer **manufacturing** *plant* and adjacent ...
B	discovered at a St. Louis *plant* **manufacturing**
B	... copper **manufacturing** *plant* found that they
B	copper wire **manufacturing** *plant* , for example ...
B	's cement **manufacturing** *plant* in Alpena ...
B	polystyrene **manufacturing** *plant* at its Dow ...
B	company **manufacturing** *plant* is in Orlando ...

Resultant Decision List

Final decision list for *plant* (abbreviated)

LogL	Collocation	Sense
10.12	*plant* growth	$\Rightarrow$ A
9.68	car (within $\pm k$ words)	$\Rightarrow$ B
9.64	*plant* height	$\Rightarrow$ A
9.61	union (within $\pm k$ words)	$\Rightarrow$ B
9.54	equipment (within $\pm k$ words)	$\Rightarrow$ B
9.51	assembly *plant*	$\Rightarrow$ B
9.50	nuclear *plant*	$\Rightarrow$ B
9.31	flower (within $\pm k$ words)	$\Rightarrow$ A
9.24	job (within $\pm k$ words)	$\Rightarrow$ B
9.03	fruit (within $\pm k$ words)	$\Rightarrow$ A
9.02	*plant* species	$\Rightarrow$ A
...	...	

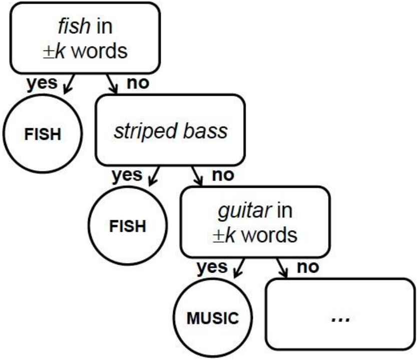
Classification of a test sentence is based on the highest ranking collocation, found in the test sentences.

plucking flowers affects plant growth.

Decision List: Example

Example: discriminating between bass (fish) and bass (music):

Context	Sense
fish in $\pm k$ words	FISH
striped bass	FISH
guitar in $\pm k$ words	MUSIC
bass player	MUSIC
piano in $\pm k$ words	MUSIC
sea bass	FISH
play bass	MUSIC
river in $\pm k$ words	FISH
on bass	MUSIC
bass are	FISH



Minimally Supervised WSD - Yarowsky

- Annotations are expensive!
- Two powerful properties of human language
  - **One Sense per Discourse**: The sense of a target word is highly consistent within any given document.
  - **One Sense per Collocation**: Nearby words provide strong and consistent clues to the sense of a target word, conditional on relative distance, order and syntactic relationship.

Yarowsky's Method

"Bootstrapping" or co-training

- Start with (small) seed, learn decision list
- Use decision list to label rest of corpus
- Retain 'confident' labels, treat as annotated data to learn new decision list
- Repeat *. . .*

Yarowsky’s Method

Example

- Disambiguating plant (industrial sense) vs. plant (living thing sense)
- Think of seed features for each sense
  - Industrial sense: co-occurring with ‘manufacturing’
  - Living thing sense: co-occurring with ‘life’
- Use ‘one sense per collocation’ to build initial decision list classifier
- Treat results (having high probability) as annotated data, train new decision list classifier, iterate

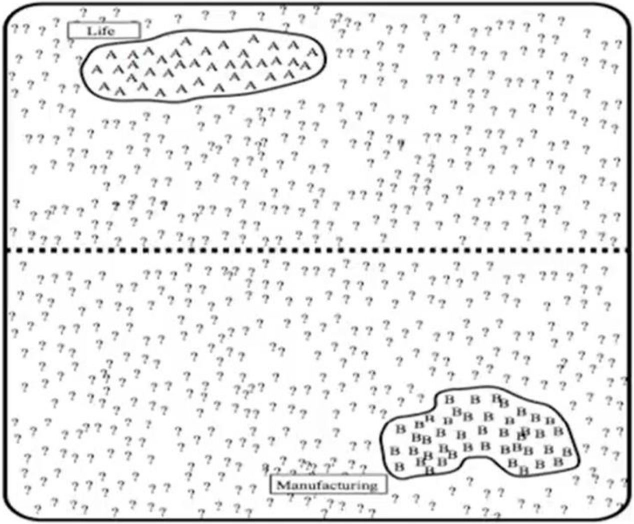
Yarowsky's Method: Example

used to strain microscopic plant life from the
zonal distribution of plant life .
close-up studies of plant life and natural
too rapid growth of aquatic plant life in water
the proliferation of plant and animal life
establishment phase of the plant virus life cycle
that divide life into plant and animal kingdom
many dangers to plant and animal life
mammals . Animal and plant life are delicately

automated manufacturing plant in Fremont
vast manufacturing plant and distribution
chemical manufacturing plant , producing viscose
keep a manufacturing plant profitable without
computer manufacturing plant and adjacent
discovered at a St. Louis plant manufacturing
copper manufacturing plant found that they
copper wire manufacturing plant , for example
s cement manufacturing plant in Alpena

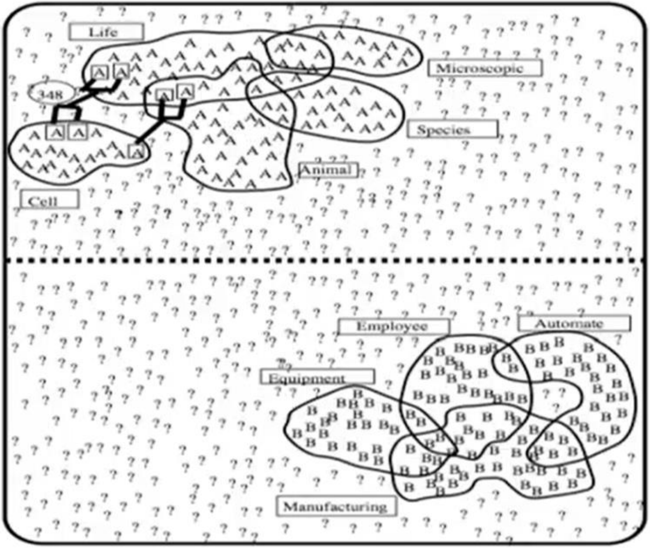
vinyl chloride monomer plant , which is
molecules found in plant and animal tissue
Nissan car and truck plant in Japan is
and Golgi apparatus of plant and animal cells
union responses to plant closures .
cell types found in the plant kingdom are
company said the plant is still operating
Although thousands of plant and animal species
animal rather than plant tissues can be

Yarowsky's Method: Example



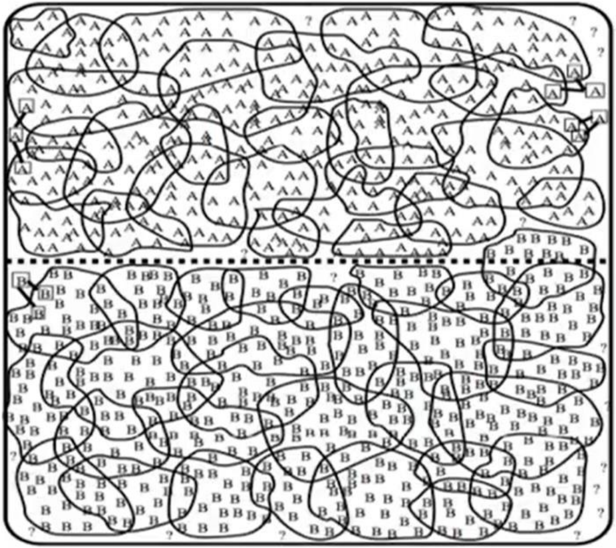
Initial state after use of seed rules

Yarowsky's Method: Example



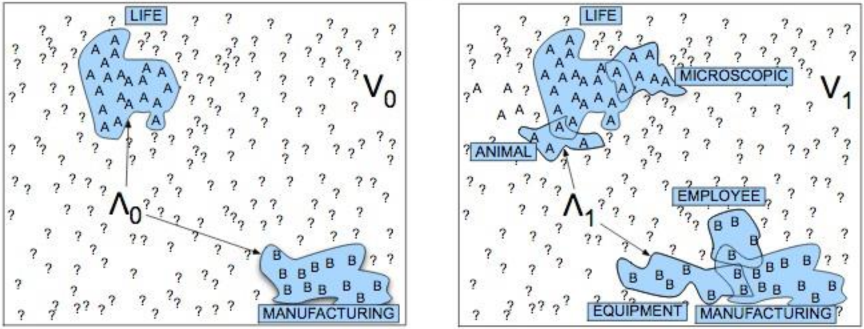
Intermediate state

Yarowsky's Method: Example



Final state

Yarowsky’s Method: Example



(a) (b)

Yarowsky's Method

Termination

- Stop when
  - Error on training data is less than a threshold
  - No more training data is covered
- Use final decision list for WSD

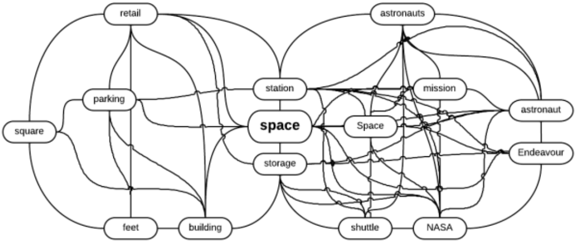
Advantages

- Accuracy is about as good as a supervised algorithm
- Bootstrapping: far less manual effort

HyperLex

Key Idea: Word Sense Induction

- Instead of using “dictionary defined senses”, extract the “senses from the corpus” itself
- These “corpus senses” or “uses” correspond to clusters of similar contexts for a word.



HyperLex

Detecting Root Hubs

- Different uses of a target word form highly interconnected bundles (or high density components)
- In each high density component one of the nodes (hub) has a higher degree than the others.
- **Step 1:** Construct co-occurrence graph, G.
- **Step 2:** Arrange nodes in G in decreasing order of degree.
- **Step 3:** Select the node from G which has the highest degree. This node will be the hub of the first high density component.
- **Step 4:** Delete this hub and all its neighbors from G.
- **Step 5:** Repeat Step 3 and 4 to detect the hubs of other high density components

Graph Weighing

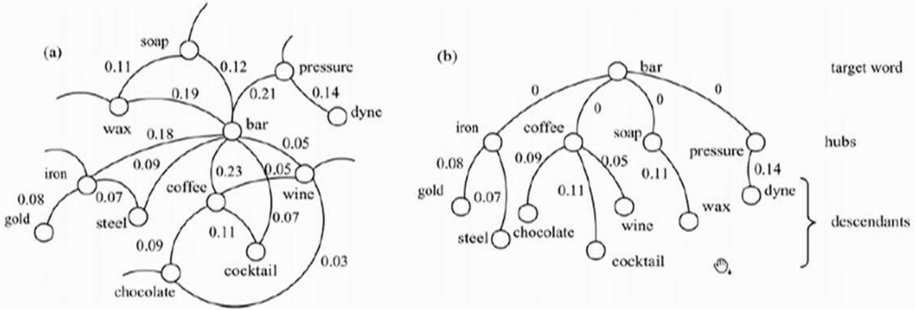
The distance between two nodes is measured as the smallest sum of weights of the edges on the paths linking them.

Computing distance between two nodes w_i and w_j

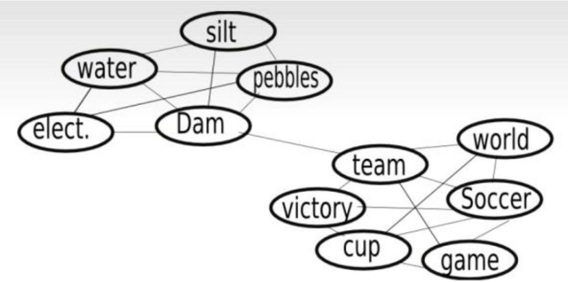
$$w_{ij} = 1 - max\{P(w_i|w_j), P(w_j|w_i)\}$$

where $P(w_i|w_j) = \frac{freq_{ij}}{freq_j}$

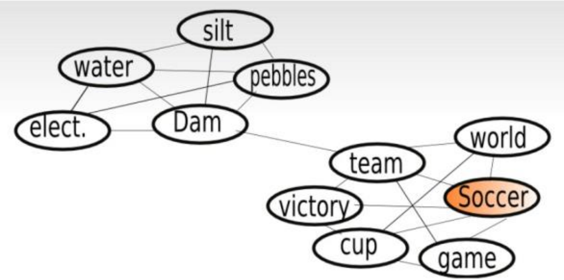
Edges with a weight above 0.9 are arbitrarily eliminated.


(a)
soap
0.11
0.12
pressure
0.19
0.21
0.14
dyne
wax
bar
0.18
0.05
iron
0.09
0.23
0.05
wine
coffee
0.08
0.07
gold
steel
0.07
0.09
0.11
cocktail
0.03
chocolate
(b)
bar
target word
0
0
0
0
iron
coffee
soap
pressure
hubs
0.08
0.09
0.05
0.11
0.14
gold
0.07
0.11
dyne
wax
wine
chocolate
steel
cocktail
descendants

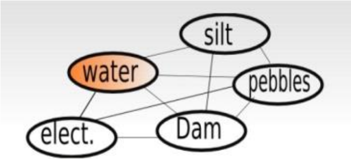
HyperLex: Detecting Root Hubs



HyperLex: Detecting Root Hubs



HyperLex: Detecting Root Hubs



Delineating Components

- **Step 1:** Once all the root hubs have been found, connect all of them with the target word with 0 edge weight in co-occurence graph.
- Identified root hubs are made the first level nodes.
- **Step 2:** Find the MST (Minimum Spanning Tree) for this graph.

Disambiguation

- Let $W = (w_1, w_2, \ldots, w_i, \ldots, w_n)$ be a context in which w_i is an instance of our target word.
- Let w_i has k hubs in its minimum spanning tree
- A score vector s is associated with each $w_j \in W (j \neq i)$, such that s_k represents the contribution of the kth hub as:

$$
\begin{aligned}
s_k &= \frac{1}{1 + d(h_k, w_j)} \text{ if } h_k \text{ is an ancestor of } w_j \\
s_i &= 0 \text{ otherwise.}
\end{aligned}
$$

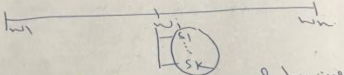
- All score vectors associated with all $w_j \in W (j \neq i)$ are summed up
- The hub which receives the maximum score is chosen as the most appropriate sense

Let $W = \{w_1, w_2 \dots w_i, \dots w_n\}$

and. Let w_i is our context word.

and w_i has k hubs in its minimum spanning tree.

Suppose we have a sentence having $w_1 \dots w_n$ words.



and w_i is target word and having sense1.... till sense k. ∴ for w_i there are k hubs.

Now for each context word $w_1, w_2 \dots w_n$ will find one score vector, each score vector is of size k.

$w_1 = [\quad]_k$

$w_2 = [\quad]_k$

$\vdots$

$w_n = [\quad]_k$

i.e. we will find one distance of each context word with all k hubs.

for e.g. $w_1 = [g - - - - - -]_k$

$1/1+d(h_1, w_1)$ or 0.

i.e. how much contribution w_1 is making to hub1, if it is ancestor than find distance $1/1+d(h_1, w_1)$ if not than 0.

Let $W = \{w_1, w_2 \ldots w_i, \ldots w_n\}$

and. Let w_i is our context word.
and w_i has k hubs in its minimum spanning tree.

Suppose we have a sentence having $w_1 \ldots w_n$ words.

and w_i is target word and having sense1 ... till sense k. ∴ for w_i there are k hubs.

Now for each context word. $w_1, w_2 \ldots w_n$ will find one score vector, each score vector is of size k.

$w_1 = [\quad]_k$

$w_2 = [\quad]_k$

⋮

$w_n = [\quad]_k$

i.e we will find one. distance of each context word. with all k hubs.

for e.g. $w_1 = [\,9 - - - - - - \,]_k$

$1/1+d(h_1,w_1)$ or 0.

i.e how much contribution w_1 is making to hub1, if it is [illegible] distance. $1/1+d(h_1,w_1)$ if not than 0.

finally will add all column wise.

$$
\begin{aligned}
w_1 &= [\ ,\ \ ,\ \ ,\ ]_k \\
w_2 &= [\ ,\ \ ,\ \ ,\ ]_k \\
&\ \ \vdots \\
w_n &= [\ ,\ \ ,\ \ ,\ ]_k \\
\hline
&\quad [\ ]\ \ [\ \ ]\ \ \ [\ ]
\end{aligned}
$$

i.e. take the sum column wise. i.e contribution of all the context words to each hubs.

and take max value.