Word Sense Disambiguation

Regina Barzilay

MIT

November, 2005

Word Sense Disambiguation

In our house, everybody has a career and none of them includes washing dishes.

I'm looking for a restaurant that serves vegetarian dishes.

- Most words have multiple senses
- Task: given a word in context, decide on its word sense

Examples (Yarowsky, 1995)

plant	living/factory
tank	vehicle/container
poach	steal/boil
palm	tree/hand
bass	fish/music
motion	legal/physical
crane	bird/machine

Harder Cases

(Some) WordNet senses of “Line”

(1) a formation of people or things one behind another

(2) length (straight or curved) without breadth or thickness; the trace of a moving point

(3) space for one line of print (one column wide and 1/14 inch deep) used to measure advertising;

(4) a fortified position (especially one marking the most forward position of troops);

(5) a slight depression in the smoothness of a surface;

(6) something (as a cord or rope) that is long and thin and flexible;

(7) the methodical process of logical reasoning;

(8) the road consisting of railroad track and roadbed;

WSD: Types of Problems

- Homonymy: meanings are unrelated (e.g., bass)
- Polysemy: related meanings (sense 2,3,6 for the word line)
- Systematic polysemy: standard methods of extending meaning

Upper bounds on Performance

Human performance indicates relative difficulty of the task

- Task: Subjects were given pairs of occurrences and had to decide whether they are instances of the same sense
- Results: agreement depends on the type of ambiguity
  - Homonyms: 95% (*bank*)
  - Polysemous words: 65% to 70% (*side, way*)

What is a word sense?

- Particular ranges of word senses have to be distinguished in many practical tasks
- There is no one way to divide the uses of a word into a set of non-overlapping categories
- (Kilgariff, 1997): senses depend on the task

WSD: Senseval Competition

- Comparison of various systems, trained and tested on the same set
- Senses are selected from WordNet
- Sense-tagged corpora available

`http://www.itri.brighton.ac.uk/events/senseval`

WSD Performance

- The accuracy depends on how difficult the disambiguation task is
  - number of senses, sense proximity, ...
- Accuracy of over 90% are reported on some of the classic, often fairly easy, WSD tasks (*interest, pike,*)
- Senseval 1 (1998)
  - Overall: 75%
  - Nouns: 80%
  - Verbs: 70%

Selectional Restrictions

- Constraints imposed by syntactic dependencies
  - *I love washing dishes*
  - *I love spicy dishes*
- Selectional restrictions may be too weak
  - *I love this dish*

Early work: semantic networks, frames, logical reasoning and “expert systems” (Hirst, 1988)

Other hints

- Single feature can provide strong evidence – no need in feature combination
- Brown et al. (1991), Resnik (1993)
  - Non-standard indicators: tense, adjacent words for collocations (*mace spray; mace and parliament*)

Automatic WSD

”You shall know the word by the company it keeps“ (Firth)

- A supervised method: decision lists
- A partially supervised method
- Unsupervised methods:
  - graph-based
  - based on distributional similarity

Supervised Methods for Word Sense Disambiguation

- Supervised sense disambiguation is very successful
- However, it requires a lot of data

Right now, there are only a half dozen teachers who can play the free bass with ease.

And it all started when fishermen decide the stripped bass in Lake Mead were too skinny.

An electric guitar and bass player stand off to one side, not really part of the scene.

Features for Word Sense Disambiguation

Right now, there are only a half dozen teachers who can play the free bass with ease.

And it all started when fishermen decide the stripped bass in Lake Mead were too skinny.

An electric guitar and bass player stand off to one side, not really part of the scene.

Features for Word Sense Disambiguation

Right now, there are only a half dozen teachers who can play the free bass with ease.

And it all started when fishermen decide the stripped bass in Lake Mead were too skinny.

An electric guitar and bass player stand off to one side, not really part of the scene.

Contextual Features in WSD

- Word found in $+/-$ k word window
- Word immediately to the right (+1 W)
- Word immediately to the left (-1 W)
- Pairs of words at offsets -2 and -1
- Pair of words at offsets -1 and +1
- Pair of words at offsets +1 and +2
- Some features are represented by their classes (WEEKDAY, MONTH)

Example

The ocean reflects the color of the sky, but even on cloudless days the color of the ocean is not a consistent blue. Phytoplankton, microscopic plant life that floats freely in the lighted surface waters, may alter the color of the water. When a great number of organisms are concentrated in an area, . . .

w_{-1} = microscopic	t_{-1}=JJ
w_{+1} = life	t_{+1}=NN
w_{-2}, w_{-1} =(Phytoplankton, microscopic)	. . .
w_{-1}, w_{+1} = (microscopic, life)	
word-within-k=ocean	
word-within-k=reflects	
. . .	

Decision Lists

- For each feature, we can get an estimate of conditional probability of sense$_1$ and sense$_2$
- Consider the feature $w_{+1} = life$:

  Count(plant$_1$, $w_{+1} = life$) =100

  Count(plant$_2$, $w_{+1} = life$) =1
- Maximum-likelihood estimate
  $P(plant_1|w_{+1} = life) = \frac{100}{101}$

Smoothed Estimates

- Problem: Counts are sparse

  Count(plant$_1$, $w_{-1} = Phytoplankton$) =2

  Count(plant$_2$, $w_{-1} = Phytoplankton$) =0

- Solution: Use α smoothing (empirically, $\alpha = 0.1$ works well):

$$P(sense\ 1\ of\ plant | w_{-1} = Phytoplankton) = \frac{2 + \alpha}{2 + 2\alpha}$$

$$P(sense\ 1\ of\ plant | w_{+1} = life) = \frac{100 + \alpha}{101 + 2\alpha}$$

  with $\alpha = 0.1$, gives values of 0.95 and 0.99 (unsmoothed gives value of 1 and 0.99)

Creating Decision Lists

- For each feature, find

$$sense(feature) = argmax_{sense} P(sense|feature)$$

e.g., $sense(w_{+1} = life) = sense_1$

- Create a rule $feature \rightarrow sense(feature)$ with weight $P(sense(feature)|feature)$

Rule	Weight
$w_{+1} = life \rightarrow plant_1$	0.99
$w_{+1} = work \rightarrow plant_2$	0.93

Creating Decision Lists

- Create a list of rules sorted by strength

Rule	Weight
$w_{+1} = life \rightarrow plant_1$	0.99
$w_{-1} = modern \rightarrow plant_2$	0.98
$w_{+1} = work \rightarrow plant_2$	0.975
word-within-k= $life \rightarrow plant_1$	0.95
$w_{-1} = assembly \rightarrow plant_2$	0.94

- To apply the decision list: take the first rule in the list which applies to an example

Applying Decision Lists

The ocean reflects the color of the sky, but even on cloudless days the color of the ocean is not a consistent blue. Phytoplankton, microscopic plant **life** that floats freely in the lighted surface waters, may alter the color of the water. When a great number of organisms are concentrated in an area, . . .

Feature	Sense	Strength
w_{-1} = microscopic	1	0.95
w_{+1} = life	1	0.99
w_{-2}, w_{-1} =	N/A	
word-within-k=reflects	2	0.65
. . .		

N/A $\rightarrow$ feature has not seen in training data

$w_{+1} = life \rightarrow Sense_1$ is chosen

Experimental Results: WSD

(Yarowsky, 1995)

- Accuracy of 95% on binary WSD

plant	living/factory
tank	vehicle/container
poach	steal/boil
palm	tree/hand

- Accent restoration in Spanish and French — 99%
  - useful for restoring accents in de-accented texts, or in automatic generation of accents while typing

Experimental Results: Accent Restoration

(Yarowsky, 1994)

- Task: to recover accents on words
  - useful for restoring accents in de-accented texts, or in automatic generation of accents while typing
  - easy to collect training/test data
- Performance: Accent restoration in Spanish and French — 99%

Automatic WSD

- A supervised method: decision lists
- A partially supervised method
- Unsupervised approaches

Beyond Supervised Methods

- If you want to be able to do WSD in the large, you need to be able to disambiguate all words in a text
- It is hard to get a large amount of annotated data for every word in a text
  - Use existing manually tagged data (SENSEVAL-2, 5000 words from Penn Treebank)
  - Use parallel bilingual data
  - Check OpenMind Word Expert project
    http://www.openmind.org/

We want unsupervised method for WSD

Local Constraints

One sense per collocation: a word reoccurring in collocation with the same word will almost surely have the same sense

- That's why decision list can make accurate predictions based on the value of just one feature

Global Constraints

One sense per discourse: the sense of a word is highly consistent within a document

- True for topic dependent words
- Not true for verbs
- Krovetz (1998): not true with respect to fine-grained senses: (e.g., language/people (English))

One sense per discourse

Tested on 37, 232 hand tagged examples

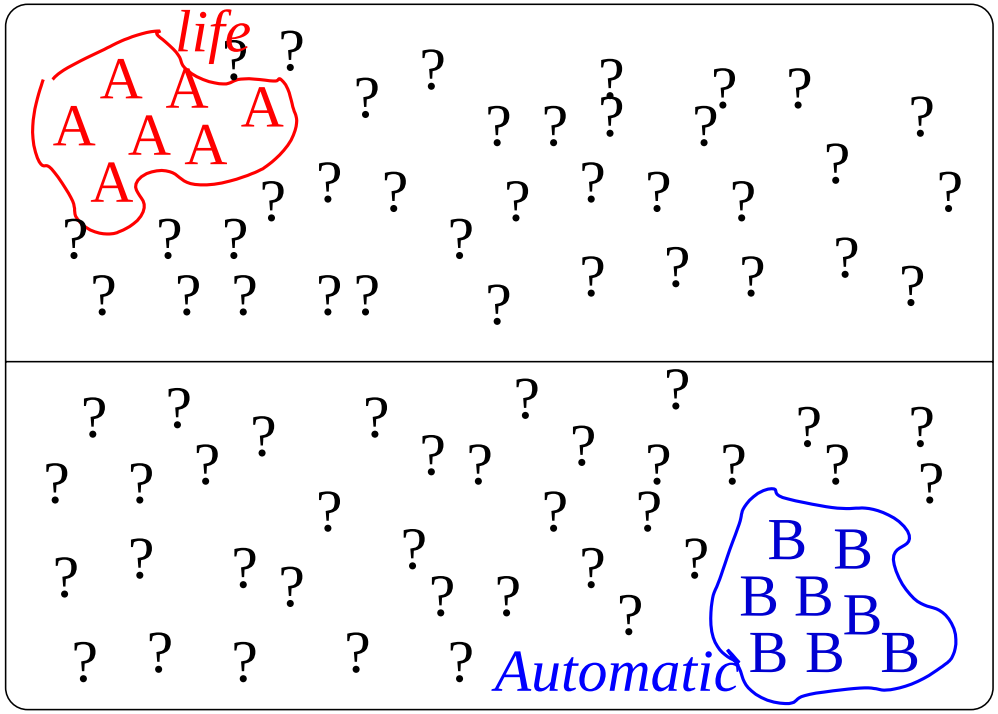
Word	Sense	Accuracy	Applicability
plant	living/factory	99.8%	72.8%
space	volume/outer	99.2%	66.2%
tank	vehicle/container	99.6%	50.5%
bass	fish/music	100.0%	58.8%
crane	bird/machine	100.0%	49.1.0

Semi-Supervised Methods

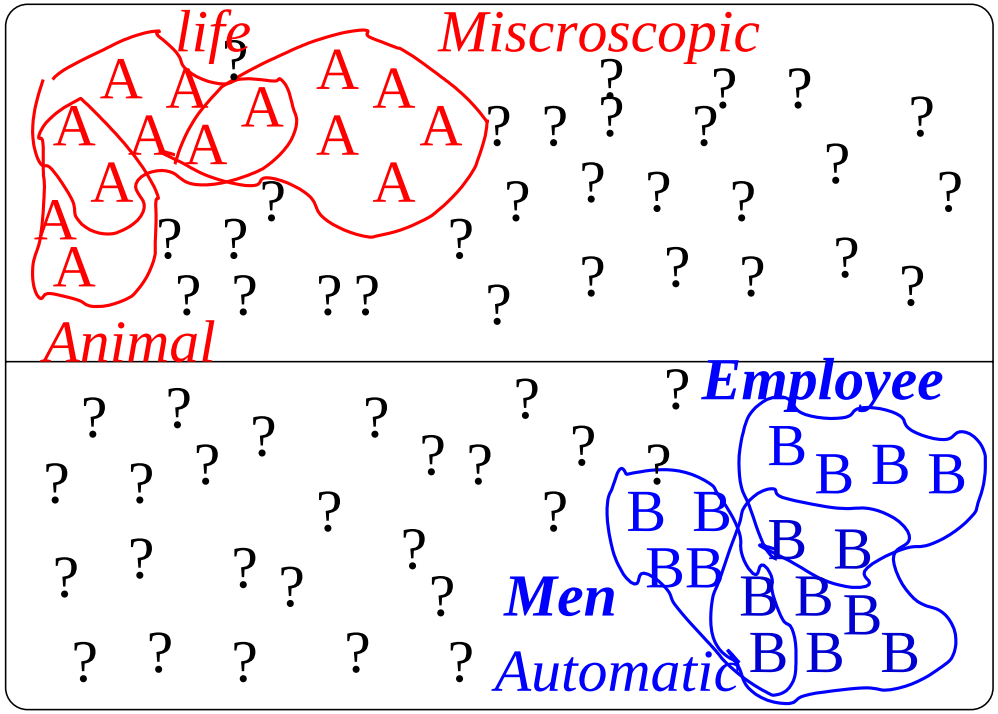
- Words can be disambiguated based on collocational features
- Words can be disambiguated based on "one sense per collocation" constraint

We can take advantage of this redundancy

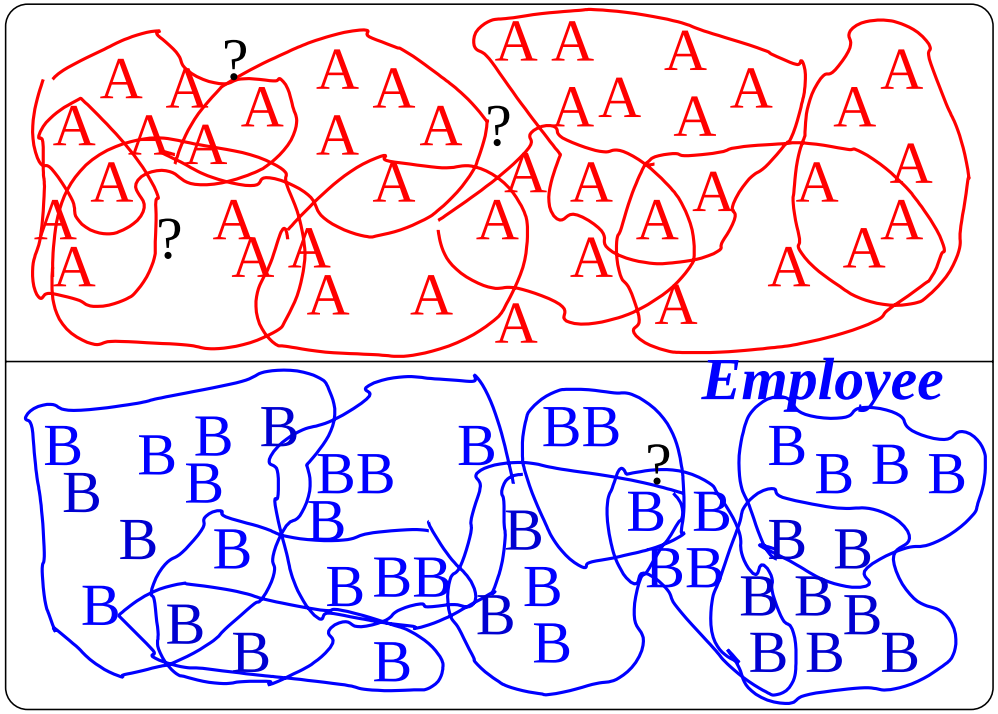
Bootstrapping Approach



Bootstrapping Approach



Bootstrapping Approach



Collecting Seed Examples

- Goal: start with a small subset of the training data being labeled
  - Label a number of training examples by hand
  - Pick a single feature for each class by hand
  - Use words in dictionary definitions
    a vegetable organism, ready for planting or lately planned
    equipment, machinery, apparatus, for industrial activity

Collecting Seed Examples: An example

- For the “plant” sense distinction, initial seeds are “word-within-k=life” and “word-within-k=manufacturing”

- Partition the unlabeled data into three sets:
  - 82 examples labeled with “life” sense
  - 106 examples labeled with “manufacturing” sense
  - 7350 unlabeled examples

Training New Rules

- From the seed data, learn a decision list of all rules with weight above some threshold (e.g., all rules with weight > 0.97)
- Using the new rules, relabel the data (thus, increasing the amount of annotated examples)
- Induce a new set of rules with weight above the threshold from the labeled data
- If some examples are still not labeled, return to step 2

Algorithm: Notations

X	set of examples, both labeled and unlabeled
Y	the current labeling
$Y^{(t)}$	the labeling at iteration t
Λ	the (current) set of labeled examples
x	an example index
j	label indices
$\perp$	unlabeled example
$\pi_x(j)$	prediction distribution
$\hat{\mathrm{y}}$	label that maximizes $\pi_x(j)$ for given x

Algorithm

(1) Given:examples X, and initial labeling $Y^{(0)}$

(2) For $t \in \{0, 1, \ldots\}$

(2.1)Train classifier on labeled examples $(\Lambda^{(t)}, Y^{(t)})$,
where $\Lambda^{(t)} = \{x \in X | Y^{(t)} \neq \perp\}$
The resulting classifier predicts label j for example x
with probability $\pi_x^{(t+1)}(j)$

(2.2)For each example $x \in X$:

(2.2.1) Set $\hat{y} = argmax_j \pi_x^{(t+1)}(j)$

(2.2.2) Set

$$Y_x^{(t+1)} = \begin{cases} Y_x^{(0)} & \text{if } x \in \Lambda^{(0)} \\ \hat{y} & \text{if } \pi_x^{(t+1)}(\hat{y}) > \zeta \\ \perp & \text{otherwise} \end{cases}$$

(2.3) If $Y^{(t+1)} = Y^{(t)}$, stop

Experiments

- Baseline score for just picking the most frequent sense for each word
- Fully supervised method
- Unsupervised Method (based on contextual clustering)

Results

Word	Sense	Samp.	Major	Superv.	Unsuperv.
plant	living/factory	7538	53.1	97.7	98.6
space	volume/outer	5745	50.7	93.9	93.6
tank	vehicle/container	11420	58.2	97.1	96.5
bass	fish/music	1859	56.1	97.8	98.8
crane	bird/machine	2145	78.0	96.6	95.5

Observations

- The results are surprisingly good
- How well does it perform on words with “weaker” sense distinctions?
- Can we predict when this method will work? (how to characterize redundancy)
- The method may not ever label all the examples

Other Applications of Co-training

- Named entity classification (Person, Company, Location)

  . . ., says Dina Katabi, an assistant professor . . .

  Spelling features: *Full-String=Dina Katabi, Contains(Dina)*

  Contextual features: *appositive=professor*

- Web page classification

  Words on the page

  Pages linking to the page

Two Assumptions Behind Co-training

- Either view is sufficient for learning

  There are functions F_1 and F_2 such that

  $$F(x) = F_1(x1) = F_2(x_2) = y$$

  for all (x, y) pairs

- Some notion of independence between the two views

  e.g.The **Conditional-independence-given-label** assumption: If $D(x_1, x_2, y)$ is the distribution over examples, then

  $$D(x_1, x_2, y) = D_0(y)D_1(x_1|y)D_2(x_2|y)$$

  for some distributions D_0, D_1 and D_2

Rote Learning, and a Graph Interpretation

- In a rote learner, functions F_1 and F_2 are look-up tables

Spelling	Category
IBM	COMPANY
Lee	PERSON
...	...

Context	Category
firm-in	LOCATION
Prof.	PERSON
...	...

- Note: no chance to learn generalizations such as "any name containing Alice is a person"

Rote Learning, and a Graph Interpretation

- Each node in the graph is a spelling or context
  (A node for IBM, Lee, firm-in, Prof.)
- Each pair (x_{1i}, x_{2i}) is an edge in the graph
  (e.g., (Prof. Lee))
- An edge between two nodes mean they have **the same label**
  (assumption 1: each view is sufficient for classification)
- As quantity of unlabeled data increases, graph becomes more connected
  (assumption 2: some independence between two views)

Automatic WSD

- A supervised method: decision lists
- A partially supervised method
- Unsupervised approaches

Graph-based WSD

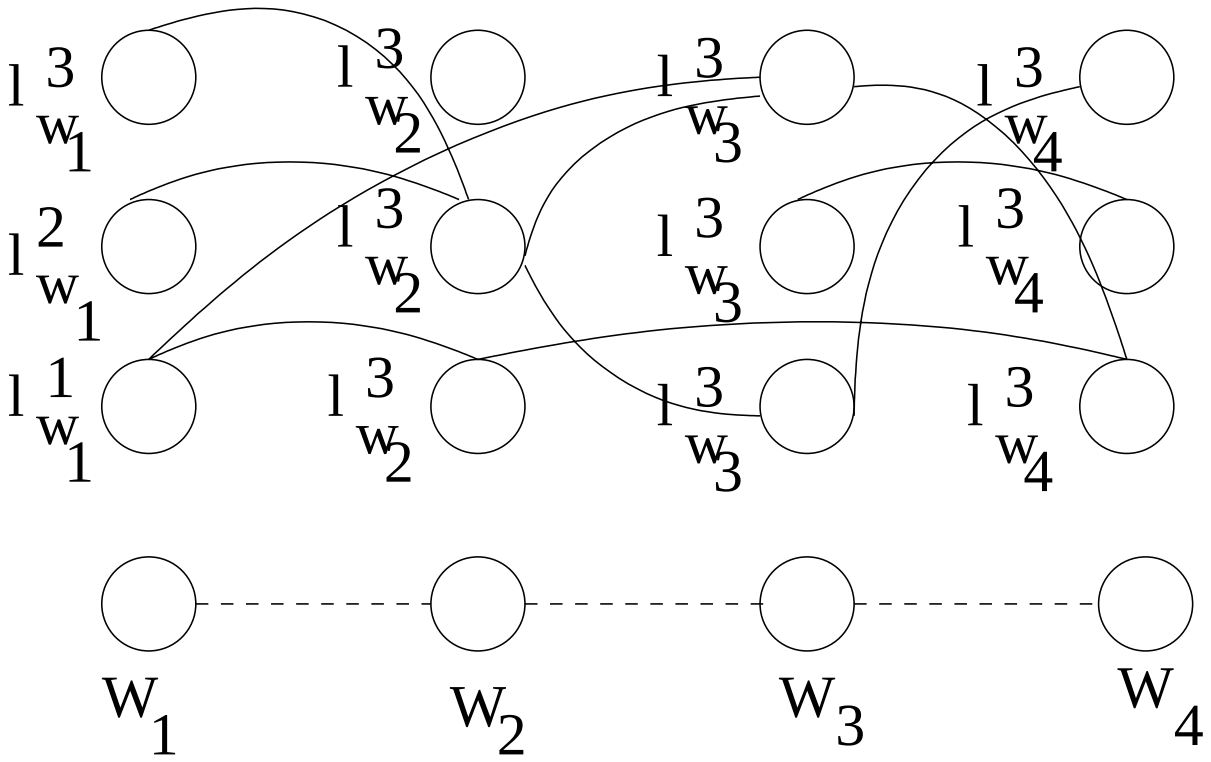
- Previous approaches disambiguate each word in isolation
- Connections between words in a sentence can help in disambiguation
- Graph is a natural way to capture connections between entities

We will apply a graph-based approach to WSD, utilizing relations between senses of various words

Graph-based Representation

- Given a sequence of words $W = \{w_1, \ldots, w_n\}$, and a set of admissible labels for each word $L_{w_i} = \{l^1_{w_i}, \ldots, l^{N_{w_i}}_{w_i}\}$
- Define a weighted graph $G(V, E)$ such that
  - V - set of nodes in the graph, where each node corresponds to a word/label assignment $l^j_{w_i}$
  - E - set of weighted edges that capture dependencies between labels

Example of Constructed Graph



Construction of Dependency Graph

for $i = 1$ to N do
  for $j = i + 1$ to N do
    for $j - i > MaxDist$ then
      break
    for $t = 1$ to N_{w_i} do
      for $s = 1$ to N_{w_j} do
        $weight \leftarrow Dependency(l^t_{w_i}, l^s_{w_i}, w_i, w_j)$
        if $weight > 0$ then
          AddEdge$(G, l^t_{w_i}, l^s_{w_i}, weight)$

Ranking Vertices and Label Assignment

$Out(v)$	out-degree of v
d	dumping factor

Vertice Ranking

repeat

  for all $v \in V$ do

    $P(v) = (1 - d) + d * \sum_{(v,q) \in E} \frac{P(q)}{Out(q)}$

until convergence of scores $P(v)$

Label Assignment

for $i = 1$ to N do

  $l_{w_i} \leftarrow$ argmax $\{P(l^t_{w_i}) | t = 1 \ldots N\}$

Computing Scores

- For data annotated with sense information, compute co-occurence statistics or sense n-grams
- For un-annotated data, compute co-occurence statistics from word glosses in WordNet

snake1	limbless scaly elongate reptile; some are venomous
snake2	a deceitful or treacherous person
crocodile1	large voracious aquatic reptile having a long snout with massive jaws and sharp teeth

Results

Random	select a sense at random
Lesk	find senses maximizing overlap in definitions

Random	37.9%
Lesk	48.7%
Graph-based	54.2%

Unsupervised Sense Reranking

- The distribution of word senses is skewed
- Selecting most common sense often produces correct results
- In WordNet, senses are ordered according to their frequency in the manually tagged SemCor
  - SemCor is small (250,000)
    for *"tiger" "audacious person"* comes before its sense as *"carnivorous animal"*
  - Most common sense is a domain-dependent notion

Automatic Sense Reranking

- Construct "distributional" cluster to which a target word belongs

  star,superstar, player, teammate

  star,galaxy, sun, world, planet

- Rank senses of the word based on the quantity and similarity of the neighbors

Cluster Construction

(Lin, 1998)

- A noun w is described by (w, r, x), where r is a grammatical relation and x is a word that co-occurs with w
- Similarity measure between w and n is computed as follows:

$$dss(w,n) = \frac{\sum_{(r,x)\in T(w)\cup T(n)} (I(w,r,x) + I(n,r,x))}{\sum_{(r,x)\in T(w)} I(w,r,x) + \sum_{(r,x)\in T(n)} I(n,r,x)},$$

where $I(w,r,x) = \log \frac{P(x|w\cup r)}{P(x|r)}$, and $T(w)$ is the set of co-occurrence types (r,x) such that $I(w,r,x) > 0$

Cluster Ranking

- Let $\{n_1, n_2, \ldots, n_k\}$ be top k neighbors with associated distributional similarity
  $M_w = \{dss(w_1, n_1), dss(w_2, n_2), \ldots, dss(w_k, n_k)\}$
- Each sense is ranked by summing over the $dss(w, n_j)$ of each neighbor multiplied by a similarity weight
- Similarity is a weight between the target sense (ws_i) and the sense of n_j that maximizes the score
  - counts number of overlapping words in glosses

Evaluation

SemCor	predominant sense from manually annotated data
SENSEVAL-2	predominant sense from the test set

	precision	recall
Automatic	64	63
SemCor	69	68
SENSEVAL-2	92	72