Pun detection

Ram manoj
Arun
Rama Chandra
Satyandhra

Motivation

- One of the central challenges in NLP.
- Ubiquitous across all languages.
- Needed in:
  - **Machine Translation**: For correct lexical choice.
  - **Information Retrieval**: Resolving ambiguity in queries.
  - **Information Extraction**: For accurate analysis of text.
- Computationally determining which ***sense*** of a word is activated by its use in a particular ***context***.
    - E.g. I am going to withdraw money from the ***bank.***
- A classification problem:
    - Senses → Classes
    - Context → Evidence

Roadmap

- **Knowledge Based Approaches**
  - WSD using Selectional Preferences (or restrictions)
  - Overlap Based Approaches
- **Machine Learning Based Approaches**
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms

KNOWLEDEGE BASED v/s MACHINE LEARNING BASED v/s HYBRID APPROACHES

- Knowledge Based Approaches
  - Rely on knowledge resources like WordNet, Thesaurus etc.
  - May use grammar rules for disambiguation.
  - May use hand coded rules for disambiguation.
- Machine Learning Based Approaches
  - Rely on corpus evidence.
  - Train a model using tagged or untagged corpus.
  - Probabilistic/Statistical models.

Roadmap

- Knowledge Based Approaches
  - Overlap Based Approaches
- **Machine Learning Based Approaches**
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms
- **Hybrid Approaches**
- **Reducing Knowledge Acquisition Bottleneck**
- **WSD and MT**
- **Summary**
- **Future Work**

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).
- These 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.

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.*

E.g. "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.

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 words 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

KB APPROACHES – COMPARISONS

Algorithm	Accuracy
WSD using Selectional Restrictions	44% on Brown Corpus
Lesk’s algorithm	50-60% on short samples of *“Pride and Prejudice”* and some *“news stories”*.
WSD using conceptual density	54% on Brown corpus.
WSD using Random Walk Algorithms	54% accuracy on SEMCOR corpus which has a baseline accuracy of 37%.
Walker’s algorithm	50% when tested on 10 highly polysemous English words.

KB APPROACHES –CONCLUSIONS

- Drawbacks of WSD using Selectional Restrictions
  - Needs exhaustive Knowledge Base.
- Drawbacks of Overlap based approaches
  - Dictionary definitions are generally very small.
  - Dictionary entries rarely take into account the distributional constraints of different word senses (e.g. selectional preferences, kinds of prepositions, etc. → c***igarette*** and ***ash*** never co-occur in a dictionary).
  - Suffer from the problem of sparse match.
  - Proper nouns are not present in a MRD. Hence these approaches fail to capture the strong clues provided by proper nouns.

E.g. **"Sachin Tendulkar"** will be a strong indicator of the category **"sports"**.

Sachin Tendulkar plays **cricket.**

Roadmap

- **Knowledge Based Approaches**
  - WSD using Selectional Preferences (or restrictions)
  - Overlap Based Approaches
- **Machine Learning Based Approaches**
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms
- **Hybrid Approaches**
- **Reducing Knowledge Acquisition Bottleneck**
- **WSD and MT**
- **Summary**
- **Future Work**

NAÏVE BAYES

$$\hat{s} = \text{argmax}_{s\,\varepsilon\,\text{senses}}\ Pr(s|V_w)$$

- 'V_w' is a feature vector consisting of:
  - POS of w
  - Semantic & Syntactic features of w
  - Collocation vector (set of words around it) → typically consists of next word(+1), next-to-next word(+2), -2, -1 & their POS's
  - Co-occurrence vector (number of times w occurs in bag of words around it)

- Applying Bayes rule and naive independence assumption

$$\hat{s} = \text{argmax}_{s\,\varepsilon\,\text{senses}}\ Pr(s).\Pi_{i=1}^{n} Pr(V_w^{\,i}|s)$$

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

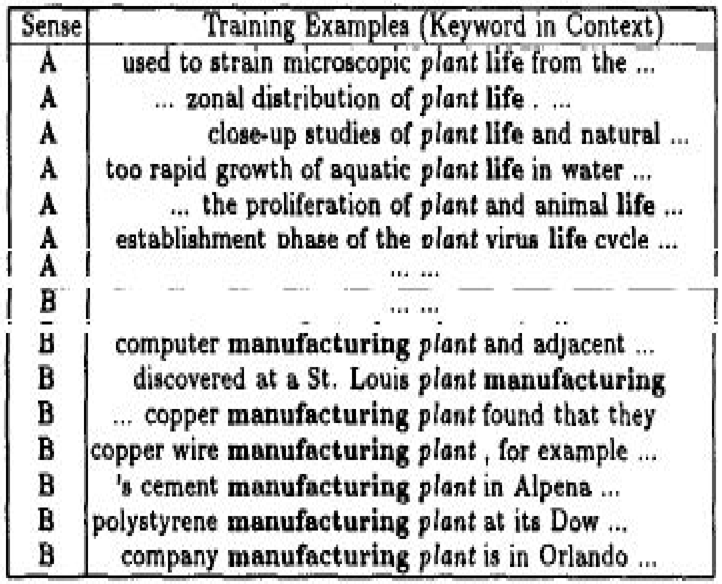
$$\mathrm{Log}\left(\frac{\Pr(\text{Sense-A}|\ \text{Collocation}_i)}{\Pr(\text{Sense-B}|\ \text{Collocation}_i)}\right)$$

Assuming there are only
two senses for the word.
Of course, this can easily
be extended to ‘k’ senses.

- Higher log-likelihood = more predictive evidence
- Collocations are ordered in a ***decision list***, with most predictive collocations ranked highest.

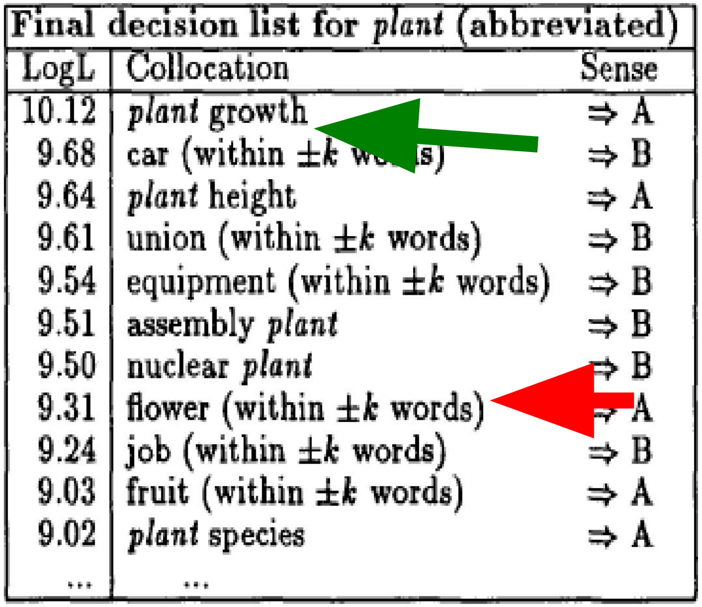
Decision List Algorithm (Contd.)

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	⇒ A
9.68	car (within ±*k* words)	⇒ B
9.64	*plant* height	⇒ A
9.61	union (within ±*k* words)	⇒ B
9.54	equipment (within ±*k* words)	⇒ B
9.51	assembly *plant*	⇒ B
9.50	nuclear *plant*	⇒ B
9.31	flower (within ±*k* words)	⇒ A
9.24	job (within ±*k* words)	⇒ B
9.03	fruit (within ±*k* words)	⇒ A
9.02	*plant* species	⇒ A
...	...	

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

E.g.

…plucking **flowers** affects ***plant growth***…

EXEMPLAR BASED WSD (K-NN)

- An exemplar based classifier is constructed for each word to be disambiguated.
- **Step1:** From each ***sense marked sentence*** containing the ambiguous word , a training example is constructed using:
  - POS of ***w*** as well as POS of neighboring words.
  - Local collocations
  - Co-occurrence vector
  - Morphological features
  - Subject-verb syntactic dependencies
- **Step2:** Given a test sentence containing the ambiguous word, a test example is similarly constructed.
- **Step3:** The test example is then compared to all training examples and the k-closest training examples are selected.
- **Step4:** The sense which is most prevalent amongst these “k” examples is then selected as the correct sense.

WSD USING SVMS

- SVM is a binary classifier which finds a hyperplane with the largest margin that separates training examples into 2 classes.
- As SVMs are binary classifiers, a separate classifier is built for each sense of the word
- **Training Phase:** Using a tagged corpus, f or every sense of the word a SVM is trained using the following features:
  - POS of ***w*** as well as POS of neighboring words.
  - Local collocations
  - Co-occurrence vector
  - Features based on syntactic relations (e.g. headword, POS of headword, voice of head word etc.)
- **Testing Phase:** Given a test sentence, a test example is constructed using the above features and fed as input to each binary classifier.
- The correct sense is selected based on the label returned by each classifier.

WSD USING PERCEPTRON TRAINED HMM

- WSD is treated as a sequence labeling task.

- The class space is reduced by using WordNet’s super senses instead of actual senses.

- A discriminative HMM is trained using the following features:
  - POS of ***w*** as well as POS of neighboring words.
  - Local collocations
  - Shape of the word and neighboring words

    E.g. for s = “Merrill Lynch & Co shape(s) =Xx*Xx*&Xx

- Lends itself well to NER as labels like “person”, location”, "time” etc are included in the super sense tag set.

SUPERVISED APPROACHES – COMPARISONS

Approach	Average Precision	Average Recall	Corpus	Average Baseline Accuracy
Naïve Bayes	64.13%	Not reported	Senseval3 – All Words Task	60.90%
Decision Lists	96%	Not applicable	Tested on a set of 12 highly polysemous English words	63.9%
Exemplar Based disambiguation (k-NN)	68.6%	Not reported	WSJ6 containing 191 content words	63.7%
SVM	72.4%	72.4%	Senseval 3 – Lexical sample task (Used for disambiguation of 57 words)	55.2%
Perceptron trained HMM	67.60	73.74%	Senseval3 – All Words Task	60.90%

SUPERVISED APPROACHES –CONCLUSIONS

- General Comments
  - Use corpus evidence instead of relying of dictionary defined senses.
  - Can capture important clues provided by proper nouns because proper nouns do appear in a corpus.
- Naïve Bayes
  - Suffers from data sparseness.
  - Since the scores are a product of probabilities, some weak features might pull down the overall score for a sense.
  - A large number of parameters need to be trained.
- Decision Lists
  - A word-specific classifier. A separate classifier needs to be trained for each word.
  - Uses the single most predictive feature which eliminates the drawback of Naïve Bayes.

SUPERVISED APPROACHES –CONCLUSIONS

- Exemplar Based K-NN
  - A word-specific classifier.
  - Will not work for unknown words which do not appear in the corpus.
  - Uses a diverse set of features (including morphological and noun-subject-verb pairs)
- SVM
  - A word-sense specific classifier.
  - Gives the highest improvement over the baseline accuracy.
  - Uses a diverse set of features.
- HMM
  - Significant in lieu of the fact that a fine distinction between the various senses of a word is not needed in tasks like MT.
  - A broad coverage classifier as the same knowledge sources can be used for all words belonging to super sense.
  - Even though the polysemy was reduced significantly there was not a comparable significant improvement in the performance.

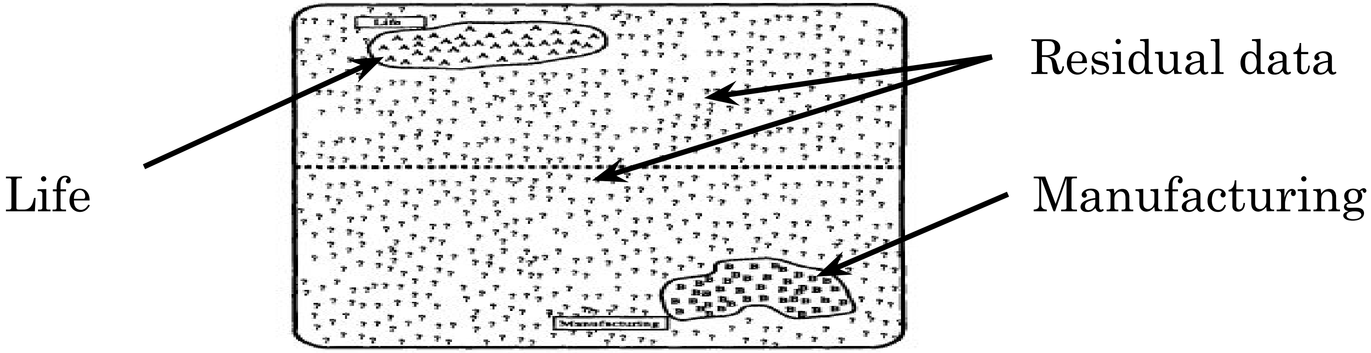
ROADMAP

- **Knowledge Based Approaches**
  - WSD using Selectional Preferences (or restrictions)
  - Overlap Based Approaches
- **Machine Learning Based Approaches**
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms
- **Hybrid Approaches**
- **Reducing Knowledge Acquisition Bottleneck**
- **WSD and MT**
- **Summary**
- **Future Work**

SEMI-SUPERVISED DECISION LIST ALGORITHM

- Based on Yarowsky's supervised algorithm that uses *Decision Lists*.
- **Step1:** Train the *Decision List* algorithm using a small amount of seed data.
- **Step2:** Classify the entire sample set using the trained classifier.
- **Step3:** Create new seed data by adding those members which are tagged as Sense-A or Sense-B with high probability.
- **Step4:** Retrain the classifier using the increased seed data.
- Exploits "One sense per discourse" property
  - Identify words that are tagged with low confidence and label them with the sense which is dominant for that document

INITIALIZATION, PROGRESS AND CONVERGENCE

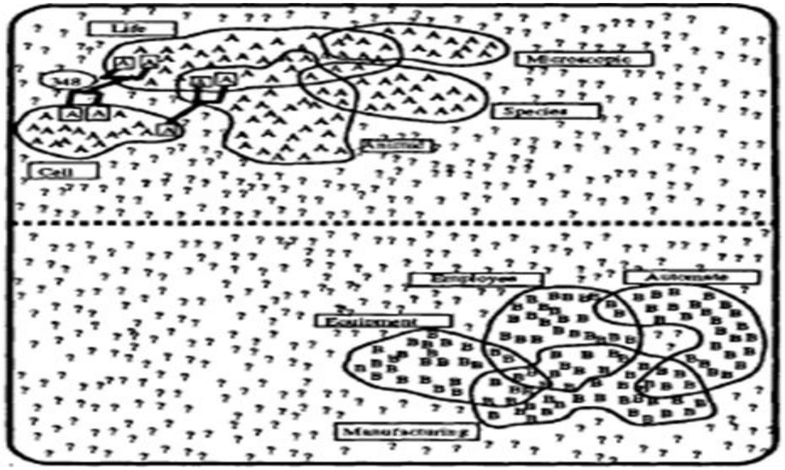


Life

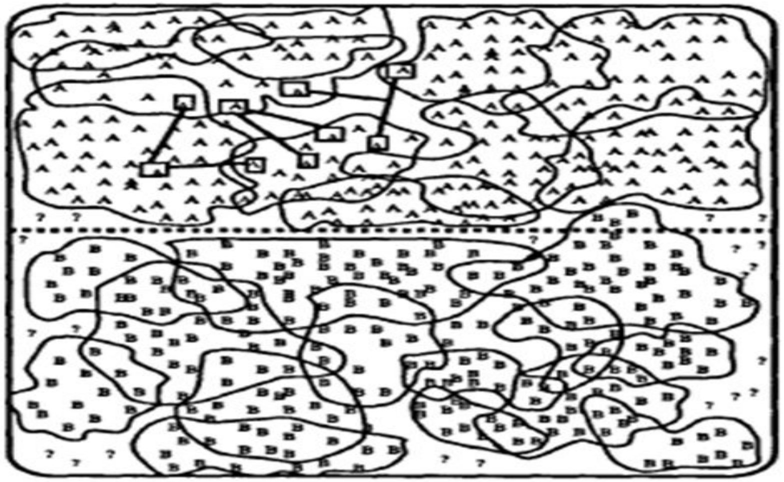
Residual data

Manufacturing

Seed set grows



Stop when residual set stabilizes



SEMI-SUPERVISED APPROACHES – COMPARISONS & CONCLUSIONS

Approach	Average Precision	Corpus	Average Baseline Accuracy
Supervised Decision Lists	96.1%	Tested on a set of 12 highly polysemous English words	63.9%
Semi-Supervised Decision Lists	96.1%	Tested on a set of 12 highly polysemous English words	63.9%

- Works at par with its supervised version even though it needs significantly less amount of tagged data.
- Has all the advantages and disadvantaged of its supervised version.

ROADMAP

- **Knowledge Based Approaches**
  - WSD using Selectional Preferences (or restrictions)
  - Overlap Based Approaches
- **Machine Learning Based Approaches**
  - Supervised Approaches
  - Semi-supervised Algorithms
  - Unsupervised Algorithms

HYPERLEX

- **KEY IDEA**
  - 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.

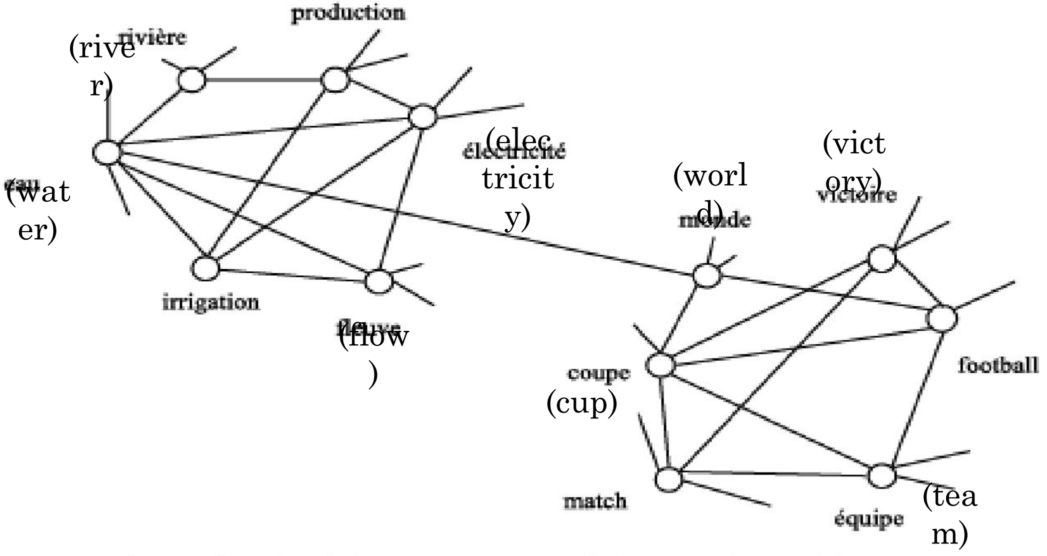
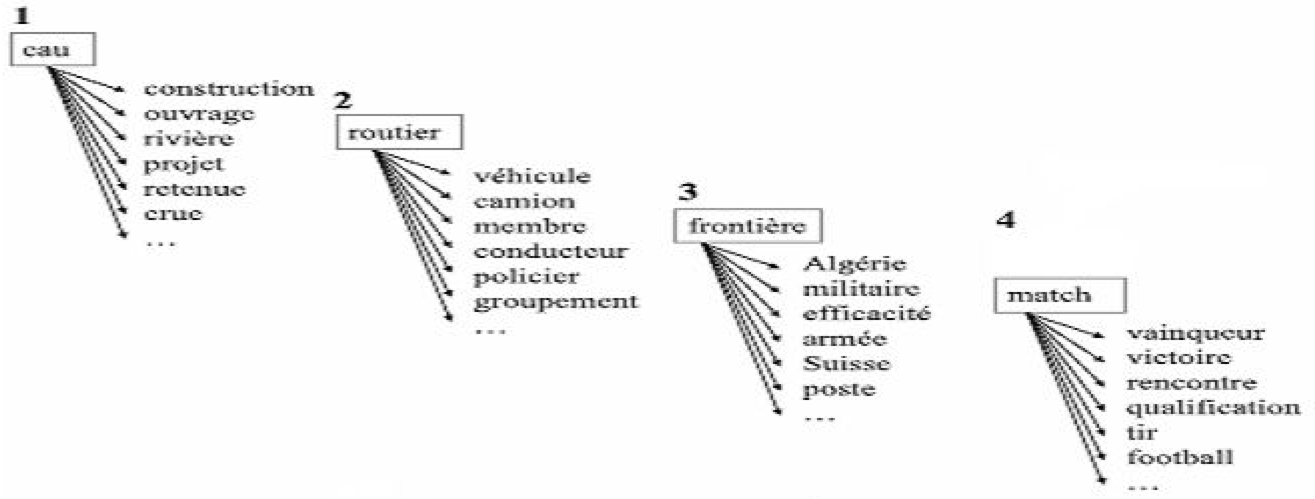


Fig. 4. Graph of the cooccurrents of the French word *barrage*.

Detecting Root Hubs

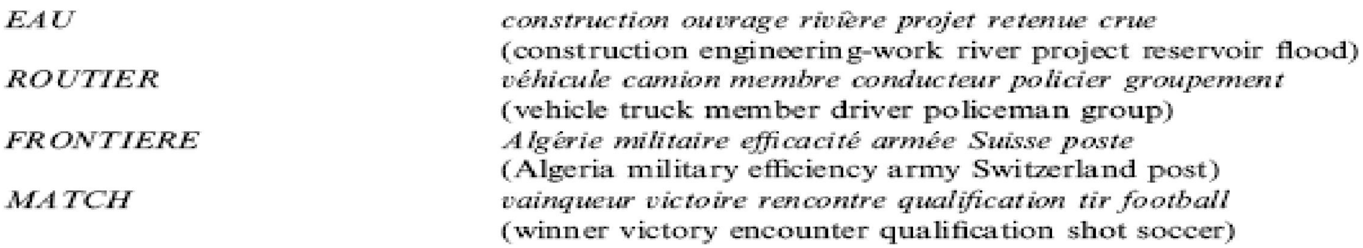
- Different uses of a target word form highly interconnected bundles (or high density components)
- In each high density component one of the nodes *(hub)* has a higher degree than the others.
- **Step 1:**
  - Construct co-occurrence graph, G.
- **Step 2:**
  - Arrange nodes in G in decreasing order of in-degree.
- **Step 3:**
  - Select the node from G which has the highest frequency. 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

DETECTING ROOT HUBS (CONTD.)



Step-by-step deletion of neighbors.

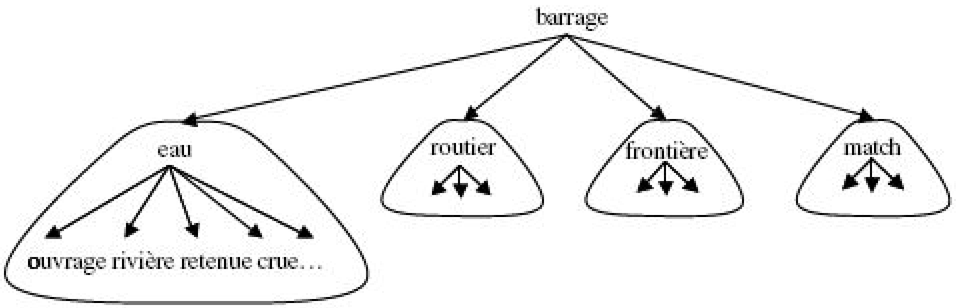
The four components for “barrage” can be characterized as:



EAU	*construction ouvrage rivière projet retenue crue* (construction engineering-work river project reservoir flood)
ROUTIER	*véhicule camion membre conducteur policier groupement* (vehicle truck member driver policeman group)
FRONTIERE	*Algérie militaire efficacité armée Suisse poste* (Algeria military efficiency army Switzerland post)
MATCH	*vainqueur victoire rencontre qualification tir football* (winner victory encounter qualification shot soccer)

DELINEATING COMPONENTS

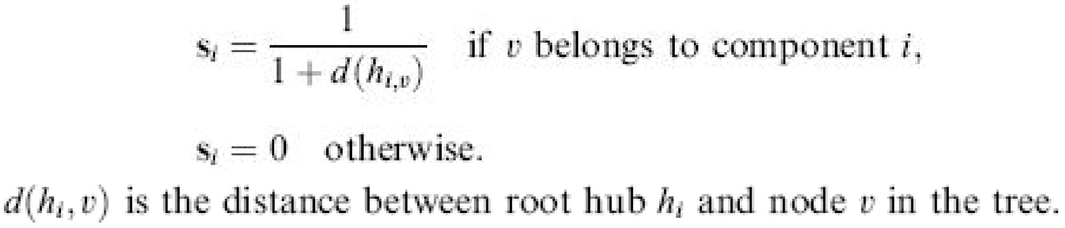
- Attach each node to the root hub closest to it.
- The distance between two nodes is measured as the smallest sum of the weights of the edges on the paths linking them.
- **<u>Step 1</u>:**
  - Add the target word to the graph G.
- **<u>Step 2</u>:**
  - Compute a *Minimum Spanning Tree (MST)* over G taking the target word as the root.



Minimum spanning tree and high-density components.

DISAMBIGUATION

- Each node in the MST is assigned a score vector with as many dimensions as there are components.



$$s_i = \frac{1}{1 + d(h_{i,v})} \quad \text{if } v \text{ belongs to component } i,$$

$$s_i = 0 \quad \text{otherwise.}$$

$d(h_i, v)$ is the distance between root hub h_i and node v in the tree.

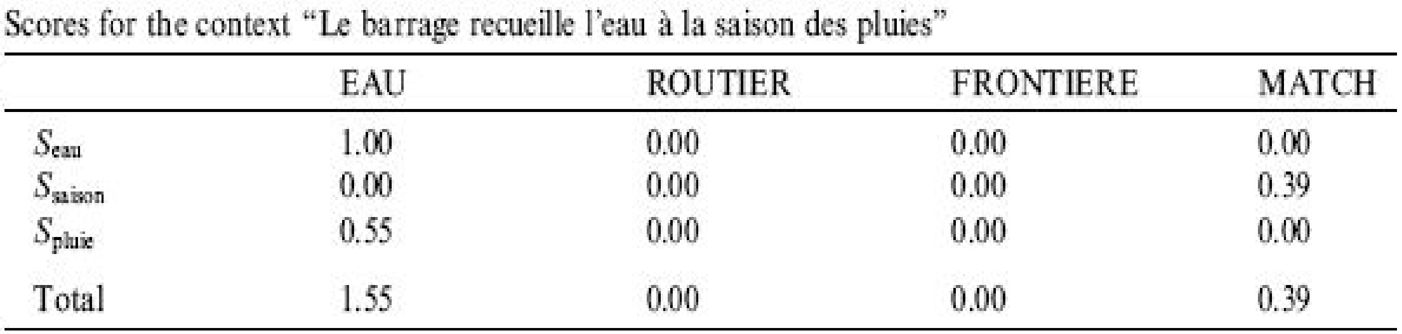
E.g. pluei(rain) belongs to the component EAU(water) and d(eau, pluie) = 0.82, s_{pluei} = (0.55, 0, 0, 0)

- **Step 1:**
  - For a given context, add the score vectors of all words in that context.
- **Step 2:**
  - Select the component that receives the highest weight.

Disambiguation (Example)

Le **barrage** recueille l’eau a la saison des plueis.

The **dam** collects water during the rainy season.



Scores for the context “Le barrage recueille l’eau à la saison des pluies”

	EAU	ROUTIER	FRONTIERE	MATCH
S_{eau}	1.00	0.00	0.00	0.00
S_{saison}	0.00	0.00	0.00	0.39
S_{pluie}	0.55	0.00	0.00	0.00
Total	1.55	0.00	0.00	0.39

EAU is the winner in this case.

A reliability coefficient (ρ) can be calculated as the difference (δ) between the best score and the second best score.

$$\boldsymbol{\rho = 1 - (1/(1+\delta))}$$

UNSUPERVISED APPROACHES – COMPARISONS

Approach	Precision	Average Recall	Corpus	Baseline
Lin's Algorithm	**68.5%.** The result was considered to be correct if the similarity between the predicted sense and actual sense was greater than 0.27	Not reported	**Trained using WSJ corpus containing 25 million words.** **Tested on 7 SemCor** files containing **2832** polysemous **nouns.**	**64.2%**
Hyperlex	**97%**	82% (words which were not tagged with confidence>threshold were left untagged)	Tested on a set of 10 highly polysemous French words	**73%**
WSD using Roget's Thesaurus categories	**92%** (average degree of polysemy was 3)	Not reported	Tested on a set of 12 highly polysemous English words	**Not reported**
WSD using parallel corpora	**SM: 62.4%** **CM: 67.2%**	SM: 61.6% CM: 65.1%	Trained using a English Spanish parallel corpus Tested using Senseval 2 – All Words task (only nouns were considered)	**Not reported**

UNSUPERVISED APPROACHES –CONCLUSIONS

- General Comments
  - Combine the advantages of supervised and knowledge based approaches.
  - Just as supervised approaches they extract evidence from corpus.
  - Just as knowledge based approaches they do not need tagged corpus.
- Lin's Algorithm
  - A general purpose broad coverage approach.
  - Can even work for words which do not appear in the corpus.
- Hyperlex
  - Use of small world properties was a first of its kind approach for automatically extracting corpus evidence.
  - A word-specific classifier.
  - The algorithm would fail to distinguish between finer senses of a word (e.g. the medicinal and narcotic senses of "drug")

UNSUPERVISED APPROACHES –CONCLUSIONS

- <u>Yarowsky’s Algorithm</u>
  - A broad coverage classifier.
  - Can be used for words which do not appear in the corpus. But it was not tested on an “all word corpus”.
- <u>WSD using Parallel Corpora</u>
  - Can distinguish even between finer senses of a word because even finer senses of a word get translated as distinct words.
  - Needs a word aligned parallel corpora which is difficult to get.
  - An exceptionally large number of parameters need to be trained.

??

THANK YOU!

??