HyperLex: lexical cartography for information retrieval

Jean Veronis

Presented by:
Siddhanth Jain(113050015)
Samiulla Shaikh(113050032)

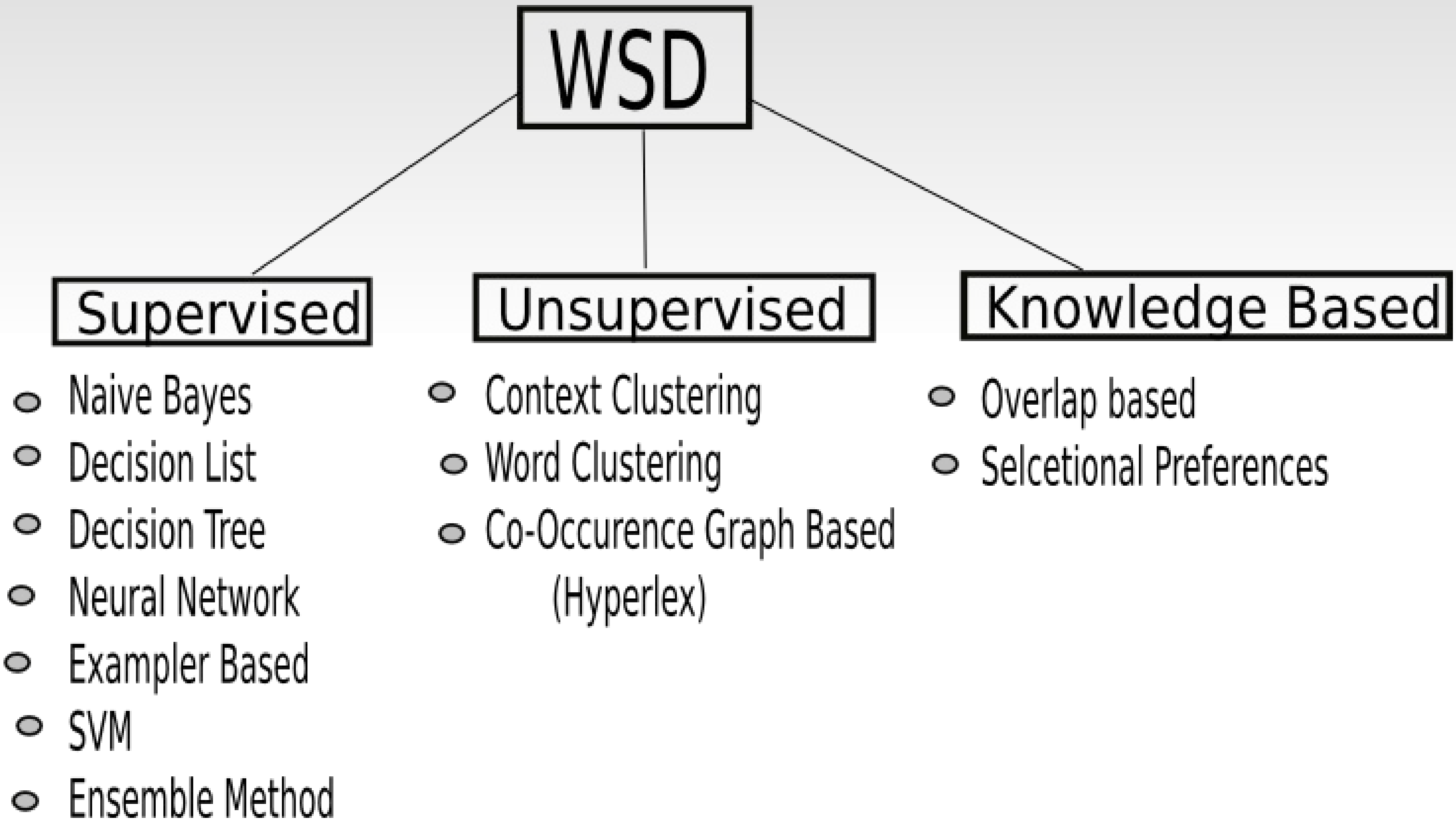
Motivation

- Human language is ambigous. Many words can have multiple meanings depending on context, domain, region etc. Such instances of words are known as polysemous.
- It is very easy to disambiguate these words for humans but for machines its a difficult job.
- Task of dismabiguating polysemous words is known as Word Sense Disambiguatio(WSD).
- WSD is one of the most fundamental problems in the field of NLP.

Motivation Contd...

- Supervised approaches to WSD give high accuracy.
- They need large amount of training data.
- Many languages and domains lack such kind of data.
- Hence semi-supervised and unsupervised approaches are also emerging as prominent options for WSD.

Various approaches to WSD



HyperLex

- Hyperlex is one of the most famous unsupervised approach for Word Sense Disambiguation.
- HyperLex is capable of auto-matically determining the uses of a word in a textbase without recourse to a dictionary.
- Despite of being unsupervised it has been found to be comparable to state-of-the-art supervised approaches.

Terminology

- **Co-occurence Graphs**

  For each target word we take all the words that co-occure with it and treat them as nodes.

- We create an edged between two nodes, A and B if their corresponding words co-occur with each other.

Terminology Contd..

- **Small World Graph**

  A small-world graph is a type of mathematical graph in which most nodes are not neighbors of one another, but most nodes can be reached from every other by a small number of hops or steps.

- Milgram (1967), was the first who proposed theterm ‘‘small world’’: any individual on the planet is only ‘‘six degrees away’’ from any other individual in the graph of social relations, even though there are several billion inhabitants.

Assigning weights to edges

- The weight that we assign to each edge reflects the magnitude of the 'semantic distance' between words:

- When w=0, the words always co-occured
- When w=1, the words never co-occured

Assigning Weights

- Each edge is assigned a weight that decreases as the association frequency of the words increases:

$$W_{AB} = 1 - max\left[p(A|B), p(B|A)\right]$$

- Where $p(A|B)$ is the conditional probability of observing A in a given context, knowing that context contains B and vice versa.

Example

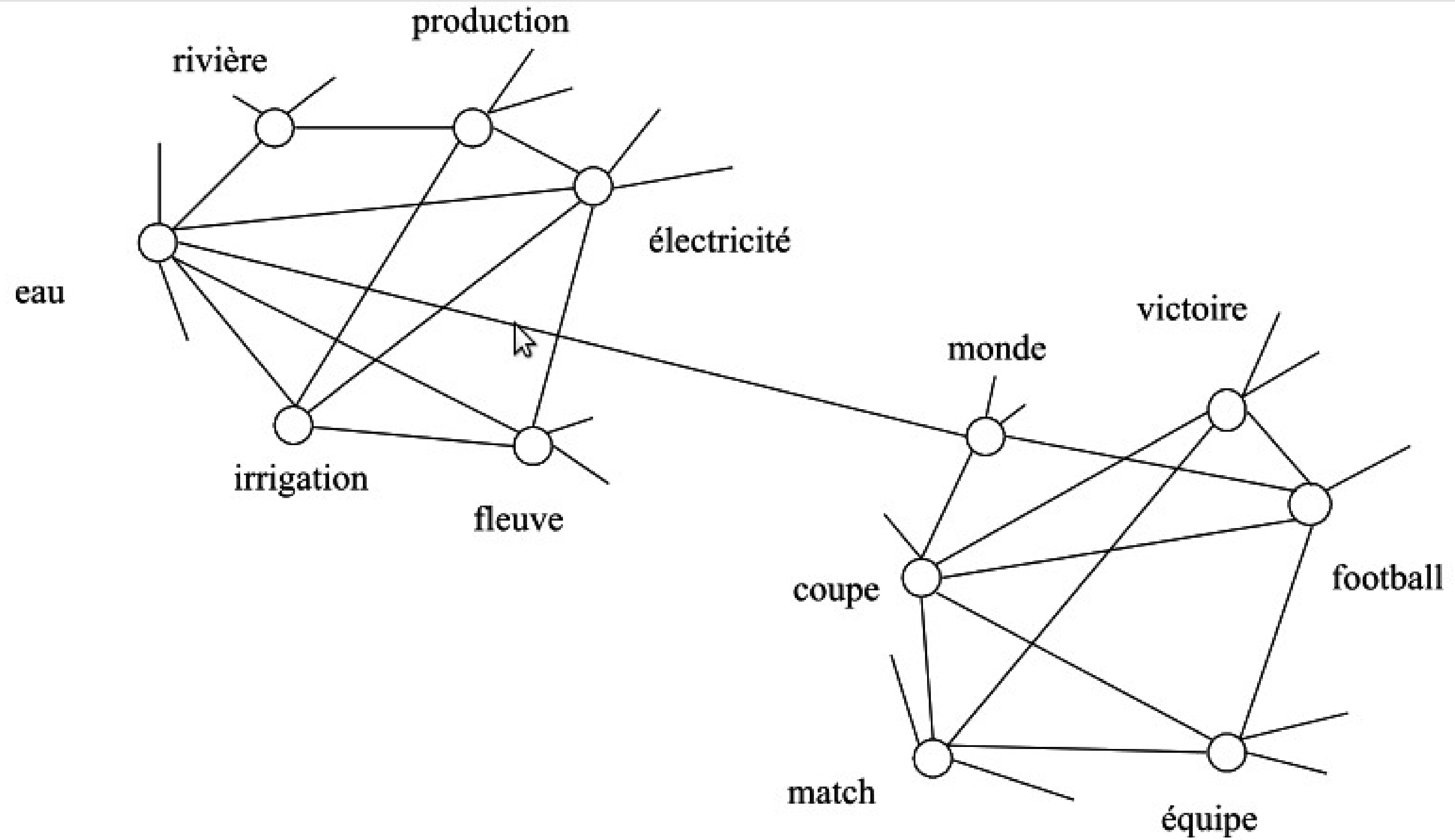
	Dam	~Dam	Total
Water	183	296	479
~Water	874	5556	6430
Total	1057	5852	6909

$$p(dam|water) = 183/479 = 0.38$$

$$w = 1 - 0.38 = 0.62$$

$$p(water|dam) = 183/1057 = 0.17$$

Co-Occurence Graph



Finding Connected Components

- Detecting the different uses of a word thus amounts to isolating the high-density components in its cooccurrence graph. Unfortunately, most exact graph-partitioning techniques are NP-hard.

- The author has given an approximate algorithm which gives fairly good results.

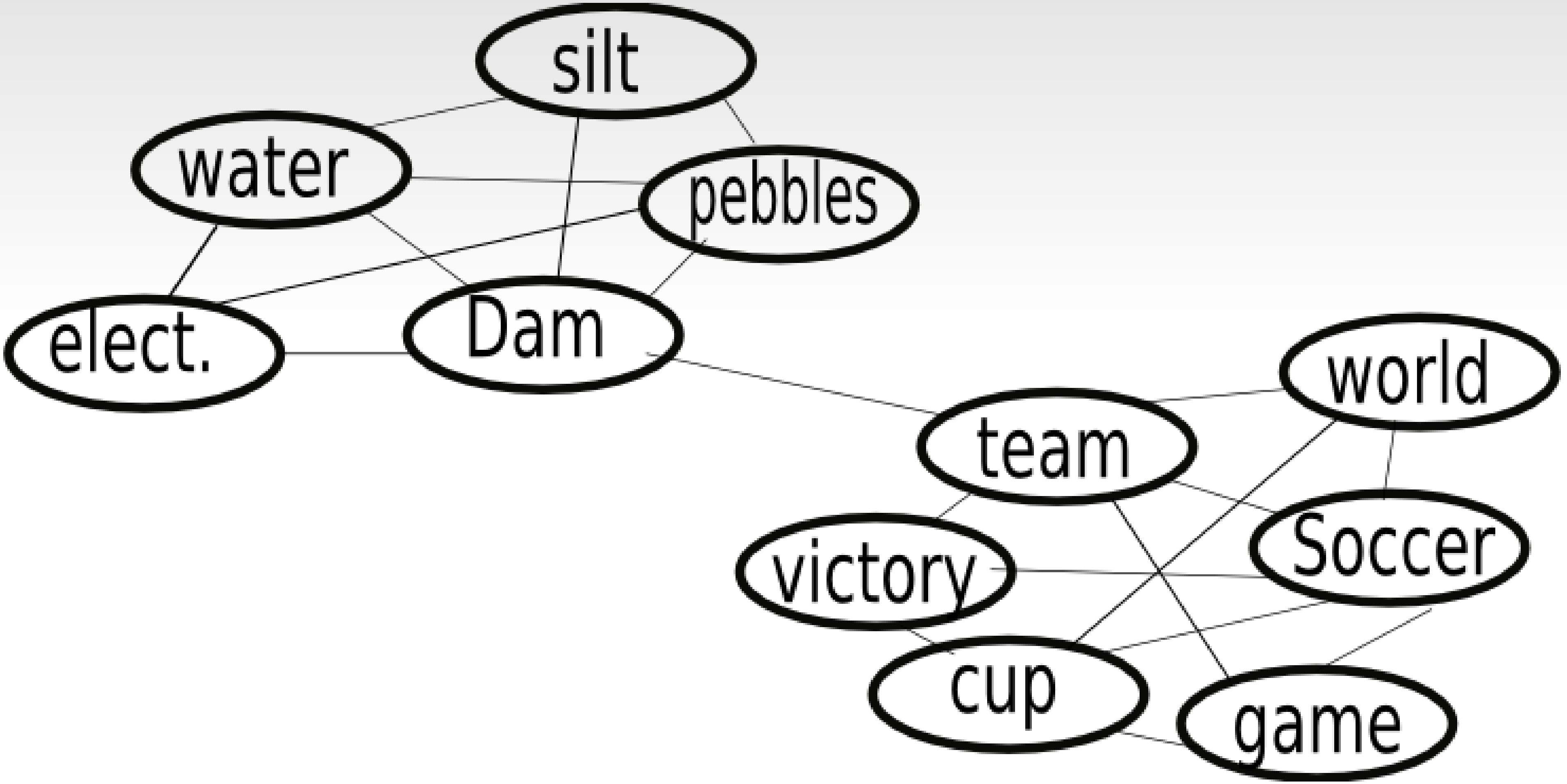
Root Hub

- In every high-density component, one of the nodes has a higher degree than the others; it is called the component's root hub.
- For example, for the most frequent use of *bank*, the root hub is the word *money*.
- It is easy to find, since it is the hub with the highest degree in the graph (and it is also the most frequent word).

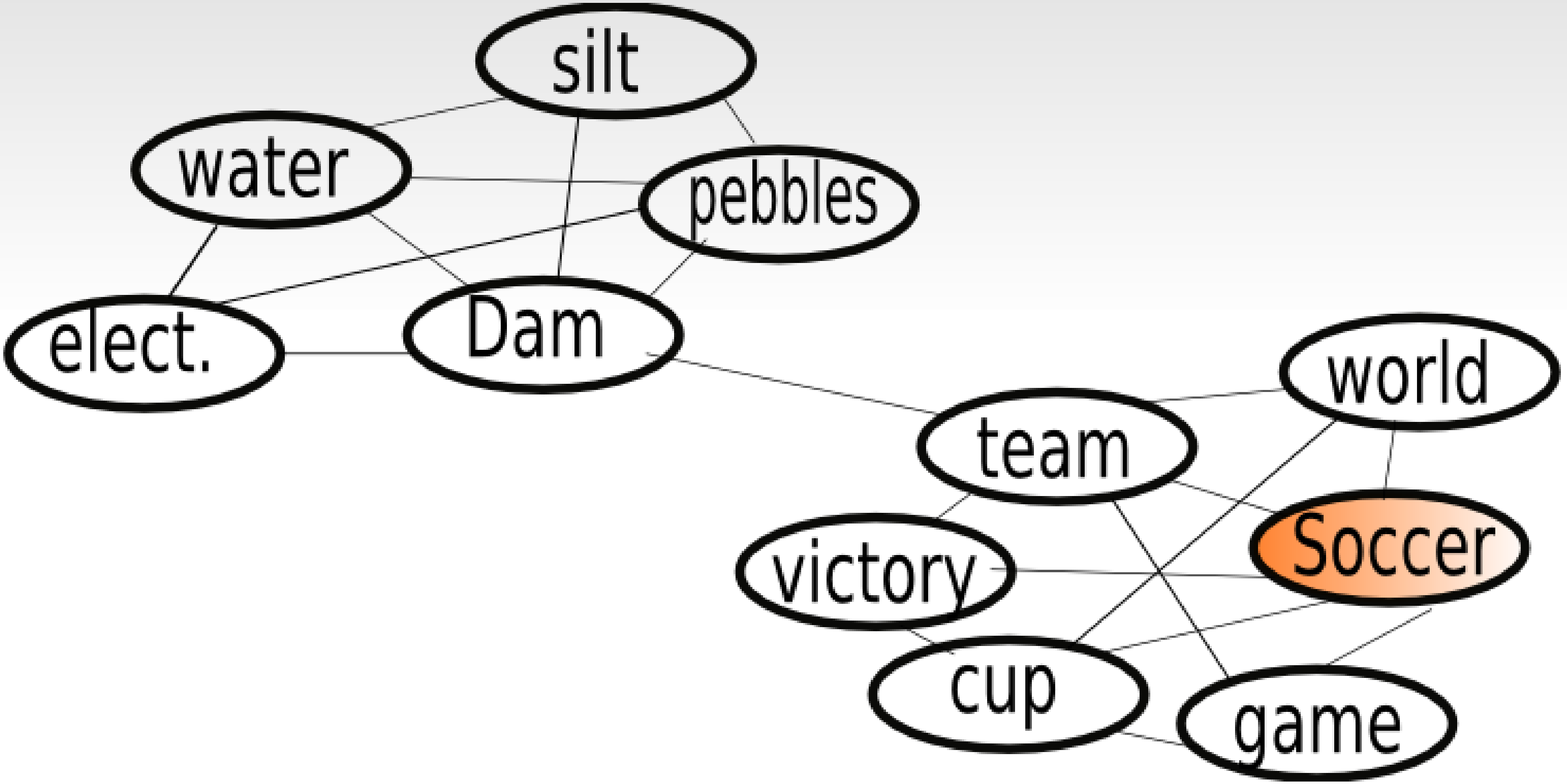
Detecting Root Hubs

- First find the highest degree node and call it the first root hub.
- Now delete the selected root hub along with all its neighbours.
- Repeat this process untill either all nodes have been covered or there is no elligible vertex for a root hub.
- A vertex is considered to be eligible for being a root hub if the 6 most frequent neighbours have weights less than 0.8(found experimentally)

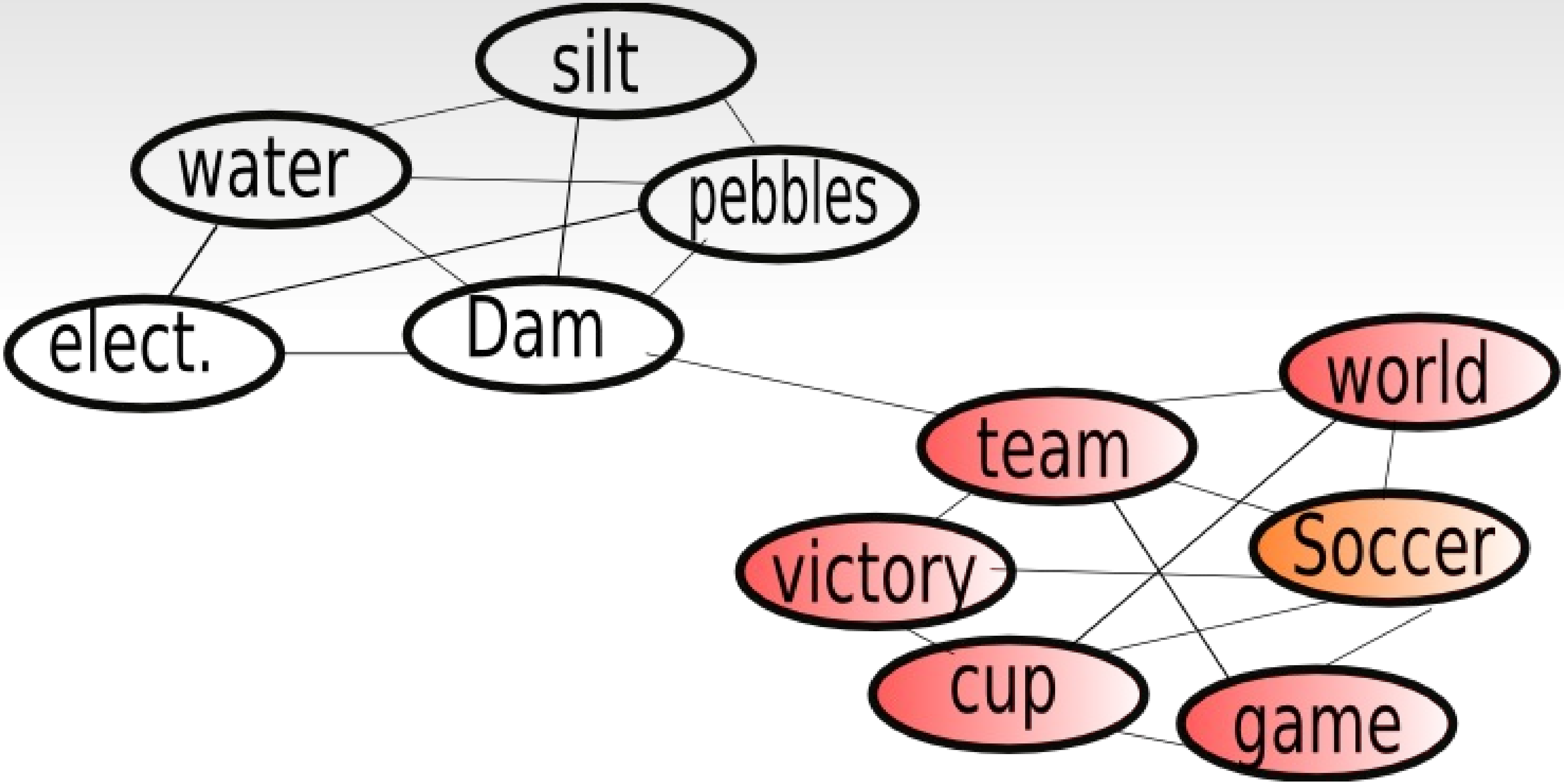
Co-Occurence Graph



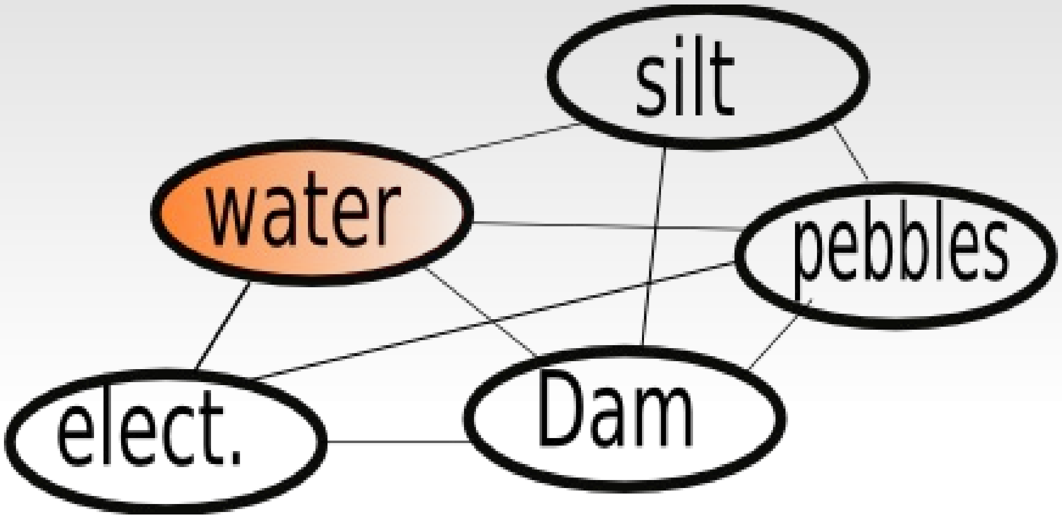
First Root Hub Detected



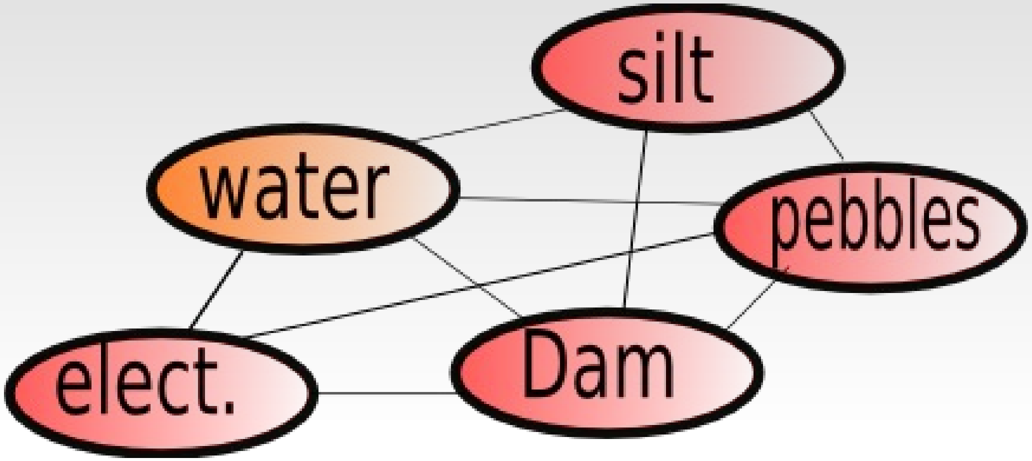
Neighbours Identified



Second Root Hub Identified from remaining graph



Neighbours Identified

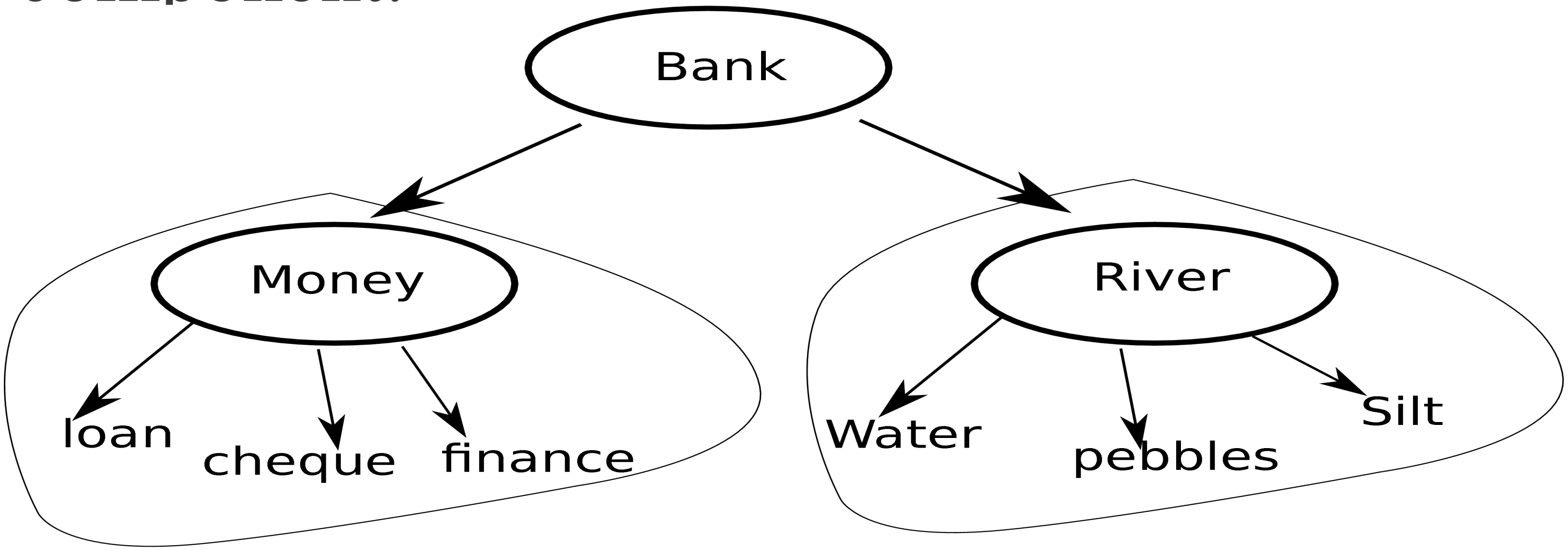


Delineating Components

- Once all the root hubs have been found connect all of them with the target word with 0 edge weight in co-occurence graph.
- Now find the MST(Minimum Spanning Tree) for this graph.

MST

- Target word is assigned a level 0.
- All root hubs are assigned a level 1.
- All nodes at level 1 represents different senses of the target word and each one of them represents a component.



Assigning Scores

- Each node in the tree is assigned a score vector of the size of the number of components.
- $s_i = \frac{1}{1+d(h_{i,v})}$ If v belongs to component i
- $s_i = 0$ Otherwise
- Now for each node in the tree we have a score vector.For example S(water)=(0,x) x>=0

Disambiguation

- Now we add the score vectors of all vertices present in the context.
- The target word is assigned the sense corresponding to the winning root hub.

Testing Results

Test word	Precision	Baseline	Error reduc.
BARRAGE	1.00	0.77	100.0%
DETENTION	1.00	0.87	100.0%
FORMATION	1.00	1.00	n/a
LANCEMENT	1.00	0.99	100.0%
ORGANE	0.88	0.40	80.0%
PASSAGE	0.88	0.52	75.0%
RESTAURATION	1.00	0.44	100.0%
SOLUTION	0.98	0.84	87.5%
STATION	1.00	0.84	100.0%
VOL	1.00	0.62	100.0%
Total	**0.97**	**0.73**	**90.4%**

Conclusion

- Hyperlex in one of the most successful unsupervised approach for WSD.
- It doesn't need any external lexical resource for disambiguation.
- Its accuracy with small number of words is comparable to state-of-the-art supervised WSD approaches.

References

- VE´RONIS, J. 2004. Hyperlex: Lexical cartography for information retrieval. Comput. Speech Lang. 18, 3,223–252
- Navigli, R. 2009. Word Sense Disambigua-tion: a survey. ACM Computing Surveys, 41(2):1–69