WoSIT: A Word Sense Induction Toolkit for Search Result Clustering and Diversification

Daniele Vannella, Tiziano Flati and Roberto Navigli

SAPIENZA UNIVERSITÀ DI ROMA

http://lcl.uniroma1.it/wosit

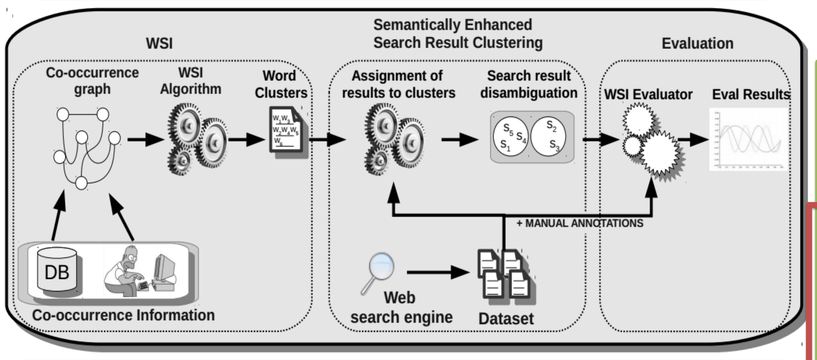
Introduction

The retrieval of any given piece of information on the Web is an arduous task which challenges even prominent search engines. It has been shown, however, that the automatic acquisition of the meanings of a word of interest (Word Sense Induction) can be successfully integrated into search result clustering and diversification so as to outperform non-semantic state-of-the-art Web clustering systems (Di Marco and Navigli, 2013).

WoSIT

We present **WoSIT**, an API for Word Sense Induction (WSI) algorithms.
The main mission of **WoSIT** is to provide a **framework** for the extrinsic evaluation of WSI algorithms.

- It provides ready implementations of existing WSI algorithms;
- It can be extended with additional WSI algorithms;
- It enables the integration of WSI algorithms into search result clustering and diversification.



WoSIT Workflow

1. WSI - Word Sense Induction;
2. Semantically-enhanced search result clustering and diversification;
3. Evaluation.

1) WoSIT - Word Sense Induction

The first phase consists of the automatic identification of the senses of a query of interest, i.e. the task of Word Sense Induction. The toolkit provides **ready-to-use implementations of several graph-based algorithms** that work with word co-occurrences. All these algorithms carry out WSI in two steps:

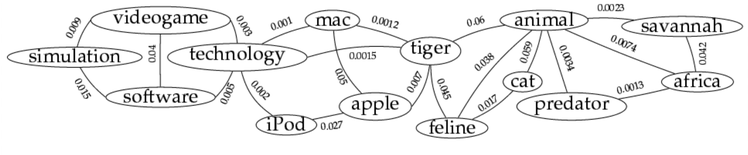
a) **co-occurrence graph construction;**

b) **discovery of word senses.**

a) Co-occurrence graph construction

Given a target query *q*, we build a **co-occurrence graph** $G_q = (V,E)$ such that *V* is the set of **words co-occurring** with *q* and *E* is the set of undirected edges, each denoting a co-occurrence between pairs of words in *V*.

Co-occurrence graph example

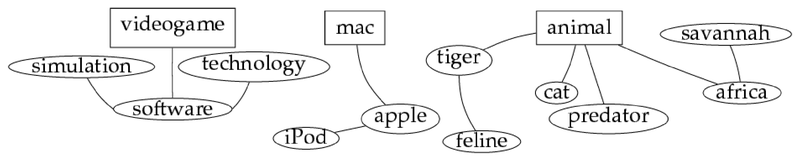


b) Discovery of Word Senses

WoSIT comes with a number of ready-to-use algorithms, among which:

- **Balanced Maximum Spanning Tree** (B-MST) (Di Marco and Navigli, 2013), an extension of a WSI algorithm based on the calculation of a Maximum Spanning Tree.
- **HyperLex** (Véronis, 2004), an algorithm which identifies hubs in co-occurrence graphs, thereby determining basic meanings for the input query.
- **Chinese Whispers** (Biemann, 2006), a randomized algorithm which partitions nodes by means of the iterative transfer of word sense information across the co-occurrence graph.
- **Squares, Triangles and Diamonds** (SquaT++), an extension of the SquaT algorithm (Navigli and Crisafulli, 2010) which exploits three cyclic graph patterns to determine and discard those vertices (or edges) with weak degree of connectivity in the graph.

Clustering example



JAVA CODE SNIPPET

```
1.  Dataset searchResults = Dataset.getInstance();
2.  DBConfiguration db = DBConfiguration.getInstance();
3.  for(String targetWord : dataset.getQueries())
4.  {
5.      WordGraph g = WordGraph.createWordGraph(targetWord, searchResults, db);
6.      BMST mst = new BMST(g);
7.      mst.makeClustering();
8.      SnippetAssociator snippetAssociator = SnippetAssociator.getInstance();
9.      SnippetClustering snippetClustering = snippetAssociator.associateSnippet(
                targetWord, searchResults, mst.getClustering(),
                AssociationMetric.WORD_OVERLAP);
10.     snippetClustering.export("output/outputMST.txt", true);
11. }
12. WSIEvaluator.evaluate(searchResults, "output/outputMST.txt");
```

3) WoSIT – Evaluation

The final component of our workflow is the **evaluation of WSI** when integrated into search result clustering and diversification (already used by Navigli and Vannella (2013)).

Two kinds of evaluations are carried out (Di Marco and Navigli, 2013):

1. Evaluation of the clustering quality
   - **Rand Index**, **Adjusted Rand Index**, **Jaccard Index**
2. Evaluation of the clustering diversity
   - **S-recall@*K***, **S-precision@*r***

The authors gratefully acknowledge the support of ERC Starting Grant MultiJEDI No. 259234.

2) WoSIT - Semantically-enhanced search result clustering and diversification

We now move to the use of the induced senses of our target query *q* within an application, i.e. search result clustering and diversification.

a) **Search result clustering;**

b) **Search result diversification.**

a) Search result clustering

It is the association of the search results returned by a search engine for query *q* with the most suitable word cluster (i.e., meaning of *q*).

Snippet clustering example

- The **Lion** King is a **video game** based on Disney's popular animated film.
- **Mac** OS X **Lion** is the eighth major release of **Mac** OS X, **Apple**'s desktop and server operating system for Macintosh ...
- For all of their roaring, growling, and ferociousness, **lions** are family animals and truly social in their own communities ...
- The **lion** is a magnificent animal that appears as a symbol of power, courage and nobility on family crests, coats of arms and national flags in many civilizations

Three different association metrics are implemented in the toolkit:

- **WORD OVERLAP** performs the association by maximizing the size of the intersection between the word sets in each snippet and the word clusters;
- **DEGREE OVERLAP** performs the association by calculating for each word cluster the sum of the vertex degrees in the co-occurrence graph of the words occurring in each snippet;
- **TOKEN OVERLAP** is similar in spirit to WORD OVERLAP, but takes into account each token occurrence in the snippet bag of words.

b) Search result diversification

The diversify method returns a flat list of snippet results obtained according to the Sorter object provided in input. The sorting rules implemented in the toolkit are **CardinalitySorter** and **MeanSimilaritySorter.**

Conclusions

- We release a **Java API** for performing Word Sense Induction which includes several ready-to-use implementations of existing algorithms;
- The API enables the use of the acquired senses for a given query for **enhancing search result clustering and diversification**;
- We provide an **evaluation component** which, given an annotated dataset of search results, carries out different kinds of evaluation of the snippet clustering quality and diversity.

References

Antonio Di Marco and Roberto Navigli. 2013. Clustering and Diversifying Web Search Results with Graph-Based Word Sense Induction. *Computational Linguistics, 39(3):709–754.*

Antonio Di Marco and Roberto Navigli. 2011. Clustering Web Search Results with Maximum Spanning Trees. *In Proc. of the XIIth International Conference of the Italian Association for Artificial Intelligence (AI*IA)*, pages 201–212, Palermo, Italy.

Jean Véronis. 2004. HyperLex: lexical cartography for information retrieval. *Computer, Speech and Language, 18(3):223–252.*

Chris Biemann. 2006. Chinese Whispers - an Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems. In *Proc. of TextGraphs: the First Workshop on Graph Based Methods for Natural Language Processing,* pages 73–80, New York City.

Roberto Navigli and Daniele Vannella.2013. SemEval-2013 Task 11: Evaluating Word Sense Induction & Disambiguation within An End-User Application. *In Proc. of the 7th International Workshop on Semantic Evaluation (SemEval 2013), in conjunction with the Second Joint Conference on Lexical and Computational Semantics (*SEM 2013),* pages 193–201, Atlanta, USA.