UoY: Graphs of Unambiguous Vertices for Word Sense Induction and Disambiguation

Ioannis Korkontzelos, Suresh Manandhar
Department of Computer Science
The University of York
Heslington, York, YO10 5NG, UK
{johnkork, suresh}@cs.york.ac.uk

Abstract

This paper presents an unsupervised graph-based method for automatic word sense induction and disambiguation. The innovative part of our method is the assignment of either a word or a word pair to each vertex of the constructed graph. Word senses are induced by clustering the constructed graph. In the disambiguation stage, each induced cluster is scored according to the number of its vertices found in the context of the target word. Our system participated in SemEval-2010 word sense induction and disambiguation task.

1 Introduction

There exists significant evidence that word sense disambiguation is important for a variety of natural language processing tasks: machine translation, information retrieval, grammatical analysis, speech and text processing (Veronis, 2004). However, the "fixed-list" of senses paradigm, where the senses of a target word is a closed list of definitions coming from a standard dictionary (Agirre et al., 2006), was long ago abandoned. The reason is that sense lists, such as WordNet (Miller, 1995), miss many senses, especially domain-specific ones (Pantel and Lin, 2002). The missing concepts are not recognised. Moreover, senses cannot be easily related to their use in context.

Word sense induction methods can be divided into vector-space models and graph based ones. In a vector-space model, each context of a target word is represented as a feature vector, e.g. frequency of cooccurring words (Katz and Giesbrecht, 2006). Context vectors are clustered and the resulting clusters represent the induced senses.

Recently, graph-based methods have been employed for word sense induction (Agirre and Soroa, 2007). Typically, graph-based methods represent each context word of the target word as a vertex. Two vertices are connected via an edge if they cooccur in one or more instances. Once the cooccurrence graph has been constructed, different graph clustering algorithms are applied to partition the graph. Each cluster (partition) consists of a set of words that are semantically related to the particular sense (Veronis, 2004). The potential advantage of graph-based methods is that they can combine both local and global cooccurrence information (Agirre et al., 2006).

Klapaftis and Manandhar (2008) presented a graph-based approach that represents pairs of words as vertices instead of single words. They claimed that single words might appear with more than one senses of the target word, while they hypothesize that a pair of words is unambiguous. Hard-clustering the graph will potentially identify less conflating senses of the target word.

In this paper, we relax the above hypothesis because in some cases a single word is unambiguous. We present a method that generates two-word vertices only when a single word vertex is unambiguous. If the word is judged as unambiguous, then it is represented as a single-word vertex. Otherwise, it is represented as a pair-of-words vertex.

The approach of Klapaftis and Manandhar (2008) achieved good results in both evaluation settings of the SemEval-2007 task. A test instance is disambiguated towards one of the induced senses if one or more pairs of words representing that sense cooccur in the test instance. This creates a sparsity problem, because a cooccurrence of two words is generally less likely than the occurrence of a single word. We expect our approach to address the data sparsity problem without conflating the induced senses.

2 Word Sense Induction

In this section we present our word sense induction and disambiguation algorithms. Figure

Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 355–358,
Uppsala, Sweden, 15-16 July 2010.

1 shows an example showing how the sense induction algorithm works: The left side of part I shows the context nouns of four snippets containing the target noun "chip". The most relevant of these nouns are represented as single word vertices (part II). Note that "customer" was not judged to be significantly relevant. In addition, the system introduced several vertices representing pairs of nouns. For example, note the vertex "company_potato". The set of sentences containing the context word "company" was judged as very different from the set of sentences containing "company" and "potato". Thus, our system hypothesizes that probably "company" and "company_potato" are relevant to different senses of "chip", and allows them to be clustered accordingly. Vertices whose content nouns or pairs of nouns cooccur in some snippet are connected with an edge (part III and right side of part I). Edge weights depend upon the conditional probabilities of the occurrence frequencies of the vertex contents in a large corpus, e.g. $w_{2,6}$ in part III. Hard-clustering the graph produces the induced senses of "chip": (a) potato crisp, and (b) microchip.

In the following subsections, the system is described in detail. Figure 2 shows a block diagram overview of the sense induction system. It consists of three main components: (a) corpus preprocessing, (b) graph construction, and (c) clustering.

In a number of different stages, the system uses a reference corpus to count occurrences of word or word pairs. It is chosen to be large because frequencies of words in a large corpus are more significant statistically. Ideally we would use the web or another large repository, but for the purposes of the SemEval-2010 task we used the union of all snippets of all target words.

2.1 Corpus Preprocessing

Corpus preprocessing aims to capture words that are contextually related to the target word. Initially, all snippets[1] that contain the target word are lemmatised and *PoS* tagged using the *GENIA* tagger[2]. Words that occur in a stoplist are filtered out. Instead of using all words as context, only nouns are kept, since they are more discriminative than verbs, adverbs and adjectives, that appear in a variety of different contexts.

[1] We refer to instances of the target word as snippets, since they can be either sentences or paragraphs.

[2] www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger

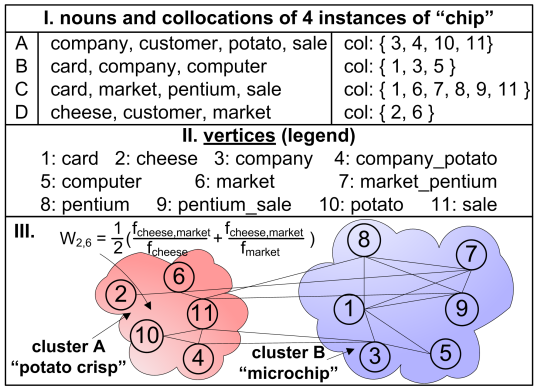

Figure 1: An example showing how the proposed word sense induction system works.

Nouns that occur infrequently in the reference corpus are removed (parameter P_1). Then, *log-likelihood ratio* (LL) (Dunning, 1993) is employed to compare the distribution of each noun to its distribution in reference corpus. The null hypothesis is that the two distributions are similar. If this is true, LL is small value and the corresponding noun is removed (parameter P_2). We also filter out nouns that are more indicative in the reference corpus than in the target word corpus; i.e. the nouns whose relative frequency in the former is larger than in the latter. At the end of this stage, each snippet is a list of lemmatised nouns contextually related to the target word.

2.2 Constructing the Graph

All nouns appearing in the list of the previous stage output are represented as graph vertices. Moreover, some vertices representing pairs of nouns are added. Each noun within a snippet is combined with every other, generating $\binom{n}{2}$ pairs. Log-likelihood filtering with respect to the reference corpus is used to filter out unimportant pairs.

Thereafter, we aim to keep only pairs that might refer to a different sense of the target word than their component nouns. For each pair we construct a vector containing the snippet IDs in which they occur. Similarly we construct a vector for each component noun. We discard a pair if its vector is very similar to both the vectors of its component nouns, otherwise we represent it as a vertex pair. Dice coefficient was used as a similarity measure and parameter P_4 as threshold value.

Edges are drawn based on cooccurrence of the corresponding vertices contents in one or more snippets. Edges whose respective vertices contents are infrequent are rejected. The weight ap-

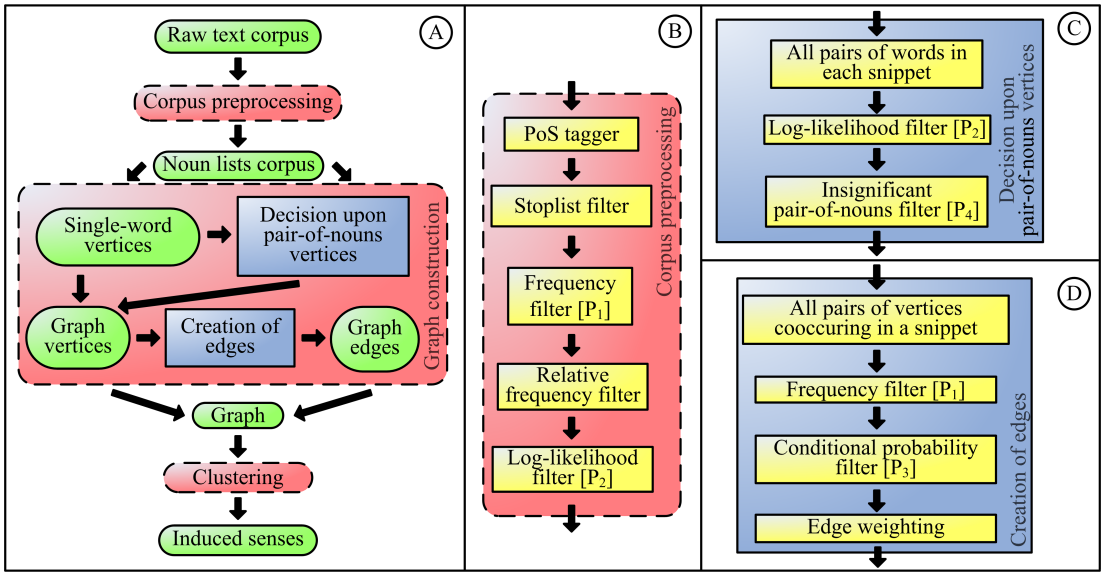

Figure 2: A: Block diagram presenting the system overview. B, C, D: Block diagrams further analysing the structure of complex components of A. Parameter names appear within square brackets.

plied to each edge is the maximum of the conditional probabilities of the corresponding vertices contents (e.g. $w_{2,6}$, part III, figure 1). Low weight edges are filtered out (parameter P_3).

2.3 Clustering the Graph

Chinese Whispers (CW) (Biemann, 2006) was used to cluster the graph. *CW* is a randomised graph-clustering algorithm, time-linear to the number of edges. The number of clusters it produces is automatically inferred. Evaluation has shown that *CW* suits well in sense induction applications, where class distributions are often highly skewed. In our experiments, *CW* produced less clusters using a constant mutation rate (5%).

To further reduce the number of induced clusters, we applied a post-processing stage, which exploits the *one sense per collocation* property (Yarowsky, 1995). For each cluster l_i, we generated the set S_i of all snippets that contain at least one vertex content of l_i. Then, any clusters l_a and l_b were merged if $S_a \subseteq S_b$ or $S_a \supseteq S_b$.

3 Word Sense Disambiguation

The induced senses are used to sense-tag each test instance of the target word (snippet). Given a snippet, each induced cluster is assigned a score equal to the number of its vertex contents (single or pairs of words) occurring in the snippet. The instance is assigned to the sense with the highest score or with equal weights to all highest scoring senses.

4 Tuning parameter and inducing senses

The algorithm depends upon 4 parameters: P_1 thresholds frequencies and P_3 collocation weights. P_2 is the LL threshold and P_4 the similarity threshold for discarding pair-of-nouns vertices.

We chose $P_1 \in \{5, 10, 15\}$, $P_2 \in \{2, 3, 4, 5, 10, 15, 25, 35\}$, $P_3 \in \{0.2, 0.3, 0.4\}$ and $P_4 \in \{0.2, 0.4, 0.6, 0.8\}$. The parameter tuning was done using the trial data of the SemEval-2010 task and on the noun data of corresponding SemEval-2007 task. Parameters were tuned by choosing the maximum supervised recall. For both data sets, the chosen parameter values were $P_1 \sim 10$, $P_3 \sim 0.4$ and $P_4 \sim 0.8$. Due to the size difference of the datasets, for the Semeval-2010 trial data $P_2 \sim 3$, while for the SemEval-2007 noun data $P_2 \sim 10$. The latter was adopted because the size of training data was announced to be large. We induced senses on the training data and then disambiguated the test data instances.

5 Evaluation results

Three different measures, V-Measure, F-Score, and supervised recall on word sense disambiguation task, were used for evaluation. V-Measure and F-Score are unsupervised. Supervised recall was measured on two different data splits. Table 1 shows the performance of our system, *UoY*, for all measures and in comparison with the best, worst and average performing system and the random and most frequent sense (MFS) baselines. Results are shown for all words, and nouns and verbs only.

	System	V-Msr	F-Sc	S-R$_{80}$	S-R$_{60}$
All	**UoY**	15.70	49.76	62.44	61.96
	Best	16.20	63.31	62.44	61.96
	Worst	0.00	16.10	18.72	18.91
	Average	6.36	48.72	54.95	54.27
	MFS	0.00	63.40	58.67	58.25
	Random	4.40	31.92	57.25	56.52
Nouns	**UoY**	20.60	38.23	59.43	58.62
	Best	20.60	57.10	59.43	58.62
	Average	7.08	44.42	47.85	46.90
	Worst	0.00	15.80	1.55	1.52
	MFS	0.00	57.00	53.22	52.45
	Random	4.20	30.40	51.45	50.21
Verbs	**UoY**	8.50	66.55	66.82	66.82
	Best	15.60	72.40	69.06	68.59
	Average	5.95	54.23	65.25	65.00
	Worst	0.10	16.40	43.76	44.23
	MFS	0.00	72.70	66.63	66.70
	Random	4.64	34.10	65.69	65.73

Table 1: Summary of results (%). V-Msr: V-Measure, F-Sc: F-Score, S-R$_X$: Supervised recall under data split: X% training, (100-X)% test

Table 2 shows the ranks of *UoY* for all evaluation categories. Our system was generally very highly ranked. It outperformed the random baseline in all cases and the MFS baseline in measures but F-Score. No participant system managed to achive higher F-Score than the MFS baseline.

The main disadvantage of the system seems to be the large number of induced senses. The reasons are data sparcity and tuning on nouns, that might have led to parameters that induce more senses. However, the system performs best among systems that produce comparable numbers of clusters. Table 3 shows the number of senses of *UoY* and the gold-standard. *UoY* produces significantly more senses than the gold-standard, especially for nouns, while for verbs figures are similar.

The system achieves low F-Scores, because this measure favours fewer induced senses. Moreover, we observe that most scores are lower for verbs than nouns. This is probably because parameters are tuned on nouns and because in general nouns appear with more senses than verbs, allowing our system to adapt better. As an overall conclusion, each evaluation measure is more or less biased towards small or large numbers of induced senses.

6 Conclusion

We presented a graph-based approach for word sense induction and disambiguation. Our approach represents as a graph vertex an unambiguous unit: (a) a single word, if it is judged as unambiguous, or (b) a pair of words, otherwise. Graph edges model the cooccurrences of the content of

	V-Msr	F-Sc	S-R$_{80}$	S-R$_{60}$
All	2	15	1	1
Nouns|Verbs	1|3	18|6	1|16	1|15

Table 2: Ranks of *UoY* (out of 26 systems)

	All	Nouns	Verbs
Gold-standard	3.79	4.46	3.12
UoY	11.54	17.32	5.76

Table 3: Number of senses

the vertices that they join. Hard-clustering the graph induces a set of senses. To disambiguate a test instance, we assign it to the induced sense whose vertices contents occur mostly in the instance. Results show that our system achieves very high recall and V-measure performance, higher than both baselines. It achieves low F-Scores due to the large number of induced senses.

References

E. Agirre and A. Soroa. 2007. Semeval-2007 task 02: Evaluating word sense induction and discrimination systems. In *proceedings of SemEval-2007*, Czech Republic. ACL.

E. Agirre, D. Martinez, O. Lopez de Lacalle, and A. Soroa. 2006. Two graph-based algorithms for state-of-the-art wsd. In *proceedings of EMNLP*, Sydney, Australia. ACL.

C. Biemann. 2006. Chinese whispers - an efficient graph clustering algorithm and its application to natural language processing problems. In *proceedings of TextGraphs*, New York City. ACL.

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

G. Katz and E. Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *proceedings of the ACL workshop on Multi-Word Expressions*, Sydney, Australia. ACL.

I. Klapaftis and S. Manandhar. 2008. Word sense induction using graphs of collocations. In *proceedings of ECAI-2008*, Patras, Greece.

G. Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

P. Pantel and D. Lin. 2002. Discovering word senses from text. In *proceedings of KDD-2002*, New York, NY, USA. ACM Press.

J. Veronis. 2004. Hyperlex: lexical cartography for information retrieval. *Computer Speech & Language*, 18(3):223–252, July.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *proceedings of ACL*, Cambridge, MA, USA. ACL.