Improved Semantic Graphs with Word Sense Disambiguation

Delia Rusu, Blaž Fortuna, Dunja Mladenić
Department of Knowledge Technologies
Jožef Stefan Institute, Ljubljana, Slovenia
{delia.rusu, blaz.fortuna, dunja.mladenic}@ijs.si

ABSTRACT
Semantic graphs can be seen as a way of representing and visualizing textual information in more structured, RDF-like graphs. The reader thus obtains an overview of the content, without having to read through the text. In building a compact semantic graph, an important step is grouping similar concepts under the same label and connecting them to external repositories. This is achieved through disambiguating word senses, in our case by assigning the sense to a concept given its context. The paper presents an unsupervised, knowledge based word sense disambiguating algorithm for linking semantic graph nodes to the WordNet vocabulary. The algorithm is integrated in the semantic graph generation pipeline, improving the semantic graph readability and conciseness. Experimental evaluation of the proposed disambiguation algorithm shows that it gives good results.

Categories and Subject Descriptors
I.2.7 [**Natural Language Processing**]: *Text analysis*.

General Terms
Algorithms, Design.

Keywords
Semantic graphs, word sense disambiguation.

1. INTRODUCTION
The majority of web content is still in the form of text, despite large efforts coming from communities such as Linking Open Data. An alternative to representing information on the Web in a textual manner is describing it semantically in the form of a graph – a semantic graph with building blocks composed of subject – predicate – object triplets automatically extracted from text sentences [5]. Based on semantic graph representation of text, a browser extension can show an overview of the current web page by displaying the semantic graph [1], making it possible for the internet surfer to get an idea regarding the web page textual content.

One issue that occurs when building semantic graphs is identifying the triplet elements that share the same meaning and can be therefore merged together in the attempt to generate a more compact graph. Additionally, connecting elements to external resources, such as WordNet, can provide useful background data which can help the understanding and the value of semantic graphs. By disambiguating word senses, only words that share the same meaning in a given context will be grouped together under the same label. The next issue is which approach to word sense disambiguation (WSD) is better suited to this scenario. While supervised WSD techniques, although they perform better than unsupervised ones, require disambiguated training data and are highly correlated to the domain of this data, unsupervised approaches offer broad coverage and can be integrated in a practical setting.

Previous work on unsupervised WSD takes the context of the target word into account; the word is disambiguated based on semantic relatedness measures [4]. In this case the context window choice is important, in order to both avoid noise induced by a window size that is too big, and include neighboring words that help in disambiguating the target word. Another approach [6] builds an acyclic weighted digraph from a target word and a number of context words, and uses the Viterbi algorithm (which ignores observation factors) to search the best path within the digraph. Our WSD algorithm builds upon the aforementioned techniques, taking advantage of similarity measures computed for word synsets within a sentence, and using the Viterbi algorithm to determine the most appropriate sequence of synsets that best reflects the meaning of a sentence.

The contribution of the paper consists of proposing an unsupervised knowledge based WSD algorithm and the way it is applied in generating more compact and readable semantic graphs.

We start by describing the word sense disambiguation algorithm in Section 2, its utility in constructing semantic graphs in Section 3, and conclude with final remarks in Section 4.

2. WORD SENSE DISAMBIGUATION
The WSD algorithm proposed in this paper is an unsupervised knowledge based one, and relies on word relatedness measures and on the Viterbi algorithm for Hidden Markov Model (HMM) tagging [2].
We compute word relatedness using WordNet::Similarity package [3] that provides measures for semantic similarity and relatedness between a pair of concepts (or synsets), all based on the WordNet lexical database. The WSD algorithm incorporates information provided by the relatedness measures (Hirst & St-Onge – hso, gloss vector, gloss vector (pairwise) and adapted Lesk (extended gloss overlaps)) which are more general, as they can be used with concepts having different parts of speech.

The Viterbi algorithm is perhaps the most common decoding algorithm used for HMMs. The version that we present takes as input a single HMM and a set of observed synsets associated to the words that we disambiguate $O = (o_1\, o_2 o_3 \ldots o_T)$, and returns the most probable state/tag sequence $Q = (q_1\, q_2 q_3 \ldots q_T)$, along with its probability.

The algorithm disambiguates word senses on a sentence level. Firstly, part of speech tagging is performed at the sentence level, and all the words are linked to their corresponding WordNet list of synsets. Secondly, we determine the sequence of observation likelihoods ($B = b_i(o_t)$), also called emission probabilities, each expressing the probability of an observation o_t being generated from a state i. For this we use WordNet's frequency count, which indicates how many times a word occured with a given synset in a tagged collection of documents. Moreover, we also generate the transition probability matrix $A = a_{11}a_{12} \dots a_{n1} \dots a_{nn}$, each a_{ij} representing the probability of moving from state i to state j. This probability is in this case the measure of relatedness between the two states (the two synsets in this case) given by the WordNet::Similarity package. In the final step of the algorithm, the Viterbi algorithm is used to find the optimal sequence of synsets, given the observation sequence and an HMM $\lambda = (A, B)$.

We evaluate the proposed algorithm on the SENSEVAL-2 corpora, containing approximately 4,000 words of text from Wall Street Journal news articles, from the Penn Treebank. Out of these, 2,260 are mapped to WordNet 3.0. In Table 1 we show the results using precision (P), recall (R) and F-measure (F). We considered two scenarios: one takes into account the frequency counts we obtained from WordNet, whereas the other does not, and assigns equal probability of occurence to all synsets. One baseline is determined by assigning the first WordNet sense, yielding an F-measure of 69% (it is common for all-words systems to be below this baseline). A lower bound can be established by randomly assigning senses to words, resulting in an F-measure of 41% (because of the words which have only one sense). Our results are comparable to the ones obtained by the best performing systems in SENSEVAL-2 all words task. The difference between using the WordNet frequency counts and assigning equal probability to observed synsets is very small.

Table 1. Evaluation results (precision, recall and F measure) for the SENSEVAL-2 all words corpora, using two relatedness measures – vector and lesk, and taking or not frequency counts provided by WordNet into account.

	Frequency Counts (%)			**No frequency counts (%)**		
S-2	P	R	F	P	R	F
vector	68.13	66.34	67.22	68.22	66.43	67.31
lesk	68.18	66.38	67.27	67.31	65.54	66.41

Another more practical experiment that we consider is taking three Reuters RCV1 newswire articles randomly (13 sentences), manually labelling them with the correct sense and measuring the precision of our algorithm. We obtain a 69.29% precision (68.57% for equal probability frequency counts) when using the *lesk* measure, and 67.86% for the *vector* measure.

3. SEMANTIC GRAPHS

The semantic graph generation pipeline, enhanced with the word sense disambiguation module, consists of the following components [5] (see Figure 1): a module for extracting named entities from text, for which co-reference and anaphora resolution are performed, followed by a triplet extraction module, where subject – predicate – object triplets are automatically identified and linked to their corresponding named entity (where available) and final module - the semantic graph generator which connects triplet elements together in a semantic graph. For these triplet elements we have determined their disambiguated word sense and merged them accordingly – the subject and object elements represent the nodes of the graph, whereas the link is labeled by the predicate.

Figure 1. The semantic graph generation pipeline enhanced with the WSD module.

4. CONCLUSIONS

This paper presented an unsupervised, knowledge based, broad coverage WSD algorithm and its application in building more compact semantic graphs. This is achieved through connecting the semantic graph elements to WordNet synsets, thus helping in better understanding the graph content.

As far as future work is concerned, we intend to expand the WSD algorithm, and evaluate the usefulness of semantic graphs, with or without the disambiguation module attached.

5. ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the IST Programme of the EC SMART (IST-033917) and PASCAL2 (IST-NoE-216886).

6. REFERENCES

[1] Dali, L., Rusu, D. and Mladenić, D. 2009 Enhanced Web Page Content Visualization with Firefox (Demo). European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - ECML PKDD.

[2] Jurafsky, D. and Martin, J. H. 2008 Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition. Prentice Hall Series in Artificial Intelligence.

[3] Pedersen, T., Patwardhan, S. and Michelizzi, J. 2004 WordNet::Similarity - Measuring the Relatedness of Concepts. In Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics, pp 38--41, Boston, MA.

[4] Patwardhan, S., Banerjee, S., and Pedersen, T. 2007 UMND1: Unsupervised Word Sense Disambiguation Using Contextual Semantic Relatedness. In Proceedings of SemEval-2007: 4th International Workshop on Semantic Evaluations, pp. 390--393, Prague.

[5] Rusu, D., Fortuna, B., Grobelnik, M. and Mladenić, D. 2009 Semantic Graphs Derived From Triplets With Application In Document Summarization. Informatica Journal, Ljubljana.

[6] Yoon, Y., Seon C-N., Lee S. and Seo, J. 2007 Unsupervised word sense disambiguation for Korean through the acyclic weighted digraph using corpus and dictionary. Information Processing and Management 43, pp 836--847.