Learning to Identify the Best Contexts for Knowledge-based WSD

Evgenia Wasserman-Pritsker  
University of Haifa  
Haifa, Israel, 31905  
evgeniaw@is.haifa.ac.il

William W. Cohen  
Carnegie Mellon University  
Pittsburgh, PA 15213  
wcohen@cs.cmu.edu

Einat Minkov  
University of Haifa  
Haifa, Israel, 31905  
einat@is.haifa.ac.il

Abstract

We outline a learning framework that aims at identifying useful contextual cues for knowledge-based word sense disambiguation. The usefulness of individual context words is evaluated based on diverse lexico-statistical and syntactic information, as well as simple word distance. Experiments using two different knowledge-based methods and benchmark datasets show significant improvements due to context modeling, beating the conventional window-based approach.

1 Introduction

Word sense disambiguation (WSD) is a key task of natural language processing. Unsupervised knowledge-based approaches to WSD (Navigli, 2009) make use of available lexical resources rather than rely on costly annotated data. Sense inference in this setting involves finding the word sense that agrees most with the specified context according to the information encoded in the knowledge base (KB). The popular Lesk (1986) method, for example, seeks to maximize word overlap between the dictionary glosses associated with the context words, and the glosses of candidate word senses. Similar methods are used in named entity disambiguation and linking to a KB (Hoffart et al., 2011).

Despite the sophistication of inference models developed, little attention has been given so far to context modeling for knowledge-based WSD. Context is represented by bag-of-words, where typically all context words are assigned equal importance (Navigli, 2009; Ling et al., 2014). However, every simple definition of context will include some unrelated or uninformative context words. Consider this usage of the word church: “An ancient stone church stands amid the fields, the sound of bells cascading from its tower”. Known senses for ‘church’ according to WordNet 3.0 (Fellbaum, 1998) correspond to a group of people, service, or a building. The latter sense is intended in this case, as one may conclude from the context words ‘stone’, ‘stands’ or ‘tower’. We wish to focus on such meaningful cues and avoid the modeling of uninformative words (‘ancient’, ‘sound’).

In this work, a learning framework is proposed that is aimed at identifying contextual cues that are predictive of the target word’s sense. The usefulness of a candidate context word for the disambiguation of the target word is evaluated based on syntactic and lexico-statistical information, as well as simple word distance. Indirect supervision is provided using noisy example labels induced automatically. Importantly, explicit lexical information is not encoded—the prediction model can thus be applied in settings where no sense-tagged examples are available of the target word type (see also (Szarsz, et al., 2013)). Having assessed the usefulness of available context words given the learned model, we consider only the top scoring context words in performing WSD.

We believe this work to be the first to perform learning-based context selection for knowledge-based sense disambiguation. Empirical evaluation using two representative knowledge-based WSD methods and different benchmark datasets indicates on consistent improvements in performance due to context selection using the proposed approach.
2 Learned context selection models (LCS)

We first define the WSD task. Given a word mention \( w \) and available context \( Ctx \), it is required to infer the intended sense \( s^* \in S(w) \), where \( S(w) \) is the set of known senses of \( w \). \( Ctx \) may be a sentence, a paragraph, or a window over words that contain \( w \).

Knowledge-based methods seek to maximize some measure of agreement, or similarity, between candidate word senses and a given context. We denote this as \( Sim() \), where the sense inference procedure is defined as follows:

\[
\hat{s}(w) = \arg \max_{s \in S(w)} Sim(s, Ctx)
\]

Typically, \( Ctx \) is represented as a bag-of-words, and the similarity score \( Sim(s, Ctx) \) is additive, i.e., it may be computed as a summation over the similarity scores between sense \( s \) and the individual context words \( c_j \in Ctx \), using the general formula:

\[
Sim(s, Ctx) = \sum_{c_j \in Ctx} weight(c_j) Sim(s, c_j)
\]

According to this view, each context word serves as a sense disambiguation ‘expert’. Context words are usually assigned uniform weight, i.e., \( weight(c_j) = 1 \). Alternatively, varying weights may reflect the reliability, or relevancy, of context word \( c_j \) in disambiguating target word \( w \); ideally, unuseful context information would be down-weighted or discarded.

2.1 Learning

Our goal is to learn models that assess whether a candidate context word \( c_j \) serves as a reliable ‘expert’ in predicting the sense of target word \( w \). We propose a distantly supervised learning scheme. Given sense-tagged instances of the form \( \langle Ctx(w_i) \rangle \), we derive a dataset of context-target word pairs \( \langle w_i, c_{ij} \rangle \), \( c_{ij} \in Ctx(w_i) \). Defining whether context work \( c_{ij} \) is useful, or reliable, with respect to the disambiguation of \( w_i \) is not trivial, however. In particular, words that are perceived as relevant according to human judgment may not necessarily yield the correct prediction using the inference algorithm. We consider a context word to be reliable if it yields a correct sense prediction of the target word, as follows:

\[
y(w_i, c_{ij}) = \begin{cases} 
1, & \text{if } \arg \max_{s \in S(w_i)} Sim(s, c_{ij}) = s^*(w_i) \\
0, & \text{otherwise}
\end{cases}
\]

As the similarity measures \( Sim() \), as well as the reference knowledge base, are imperfect, the labels assigned in this fashion are expected to be noisy.

A context and target word pair is represented as a feature vector, as described below. Importantly, we avoid the representation of explicit lexical information, so that the learned models are applicable to word pairs of arbitrary word types.

Given a new instance at test time, the learned model is used to score the individual context words. One can then assign respective non-uniform weights to the context words (Eq. 2). Here, we take a context selection approach—a ranking is induced over the context words based on the predicted scores, and only the top ranked context words are modelled in the disambiguation process; that is, the selected context words are assigned weight 1.0, and the weight of the remaining context words is set to zero. As discussed in Sec.3, this design choice was found to give preferable results in preliminary experiments.

2.2 Feature Types

Various aspects may be modeled as features in this framework, describing properties of the context word \( c_{ij} \), as well as the relationship between the target-context word pair \( \langle w, c_{ij} \rangle \). In addition to simple word distance, we encode the following syntactic and lexico-statistical information.

Syntactic features. Word distance is further assessed in syntactic terms, denoting the length of the shortest dependency path linking the word pair, as well as the length of the shortest connecting path in a constituency parse tree (Swanson and Gordon, 2006; Huang and Lu, 2011). It may be useful to further encode information about the edge types that comprise the connecting path, as some dependency relations indicate more salient semantic relatedness than others (Padó and Lapata, 2007; Minkov and Cohen, 2013). In this work, if the target and context words are directly connected in the dependency graph, we include a feature indicating the label of the edge. The part-of-speech tag of the context word may provide another contextual cue (Yarowsky, 1993); dedicated features indicate whether \( c_{ij} \) is tagged as noun, verb, adjective or adverb. We used the Stanford parser (de Marneffe et al., 2006) in our experiments.
Lexico-statistical information. We use the pointwise mutual information (PMI) measure (Turney, 2001) to assess the semantic relatedness between the context-target word pair. In general, we expect context words that are topically related to the target word to be useful for its disambiguation. To compute PMI, we obtained word frequencies from the large uKWaC corpus (Ferraresi et al., 2008), considering word co-occurrences over a window of five words. It has been indicated that highly frequent words are generally less topical, where this aspect is not fully captured by PMI (Han et al., 2013). We therefore model as complimentary information the inverse document frequency (Salton and McGill, 1983) of \(c_j\), also computed using uKWaC. Finally, a context word is often ambiguous by itself, where low polysemy is correlated with topic-specificity (Han et al., 2013). We represent the number of known senses of the context word \(c_j\) based on WordNet.

3 Experiments

We consider two WSD methods representative of prevalent knowledge-based approaches, comparing against previously published results. The popular Lesk approach (1986) mentioned before computes \(\text{Sim}(s, c_j)\) in terms of word overlap between the glosses of the senses of \(c_j\) and the gloss of \(s\). There exist multiple variants of the Lesk algorithm (Kilgarriff and Rosenzweig, 2000; Banerjee and Pedersen, 2003; Ponzetto andNavigli, 2010). We experiment with Gloss vectors (GV) (Patwardhan and Pedersen, 2006). This method enriches WordNet glosses with glosses of hypernyms and other related senses, as well as with co-occurring words derived from raw text. GV scores were obtained using the WordNet::Similarity package (Pedersen et al., 2004).

Graph-based methods are also commonly used for sense disambiguation (Mihalcea, 2005; Hughes and Ramage, 2007). If the KB is represented as a graph, various metrics can be applied that reflect structural similarity between word senses represented as graph nodes. We consider the Personalized PageRank (PPR) algorithm, which has been shown to yield state-of-the-art WSD performance (Agirre and Soroa, 2009). According to the linearity theorem (Jeh and Widom, 2003), PPR scores can be computed for each of the context words separately, and then aggregated (Eq. 2). In this case, \(\text{Sim}(s, c_j)\) equals the PPR score attributed to the node denoting sense \(s\), having the graph walk initiated at a uniform distribution over the various senses of \(c_j\). PPR scores were generated using the UKB software (Agirre and Soroa, 2009).1

3.1 Datasets

We experiment with two benchmark datasets. The lexical sample due to Koeling et al. (2005) includes annotated instances of 41 selected nouns. About 300 example sentences are available per noun, retrieved evenly from three sources: the domain-specific sports and finance sections of Reuters corpus, and the general British National Corpus (BNC). The second dataset consists of all noun examples from the SemEval-2007 English lexical sample task (Pradhan et al., 2007), created from another corpus—the WSJ Treebank.

The two datasets were transformed into target-context word pairs. For every word pair \(|w, c_j|\), the scores \(\text{Sim}(s, c_j), s \in S(w)\), were generated using GV and PPR and WordNet 3.0 as the reference knowledge base. A context-target word pair was labeled as a positive example if it yielded a correct sense prediction, or as negative otherwise (Eq. 3). Table 1 details statistics of the original and respective word pair datasets, including the ratio of context words labeled as positive examples— as shown, this ‘pairwise accuracy’ is low, reaching up to 0.35.

<table>
<thead>
<tr>
<th>Word types</th>
<th>Target words</th>
<th>Context words</th>
<th>Pairwise acc. (PPR/GV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Koeling et al.</td>
<td>41</td>
<td>9.6K</td>
<td>121K</td>
</tr>
<tr>
<td>SemEval’07</td>
<td>35</td>
<td>16K</td>
<td>390K</td>
</tr>
</tbody>
</table>

Table 1: The experimental datasets: statistics

3.2 Experimental setup

We experimented with several classification paradigms using the Weka learning suite (Hall et al., 2009). Learning had to be robust to label noise. We report results using Naive Bayes, due to both its good performance and efficiency. Following preliminary experiments, we adopted a context selection approach—the learned model is used to rank the available context words, where the top ranked words, obtained by applying ratio \(r\), are selected as context. We tune \(r\) using training examples. The reported performance uses rough values of \(r = 0.5\) for the Koeling et al. examples, which

---
1http://ixa2.si.ehu.es/ukb/; we used the bin file wn30+gloss, and the PPR.w2w graph walk variant.
include individual sentences, and \( r = 0.2 \) for SemEval’07, where paragraphs of a few sentences are provided as context.

### 3.3 Results

Table 2 shows the results of applying context selection for each of the dataset and methods. As in previous works, performance is reported in terms of recall, defined as the ratio of correct sense predictions out of total number of target word mentions.\(^2\) To avoid overfitting, we performed cross word evaluation, predicting contexts for all instances of each word type with a model trained using the other word types (LCS: CW). Concretely, the Koeling et al. dataset was split into 41 bins, according to the target word type. For each word type, we generated a model using the examples of the remaining (in this case, 40) word types. This cross word evaluation procedure was applied to both datasets. We further report the results of cross dataset experiments (LCS: CD), in which one dataset is used for training and the other for evaluation. As baseline, we use all of the available context words, weighting them uniformly (“uniform” in the table).

As shown, LCS yields substantial improvements over the “uniform” baseline. The improvement rate for each experiment is displayed in superscript. Recall increased at high rates on the SemEval dataset. This dataset is skewed, and much of these gains are attributed to large increase in recall for two word types, covering 27% of the examples. Improvements on the balanced dataset due to Koeling et al. were more modest, yet significant. Interestingly, improvement rates are higher using GV than PPR; we conjecture that PPR predictions are biased towards highly-connected graph nodes, being less sensitive to the local context defined. Remarkably, the results using cross-dataset training are comparable to or exceed the within-dataset CW results, showing generality and robustness of the proposed approach.

\(^2\)Since predictions are generated for all examples, recall equals in this case to precision, and accuracy.
hurt performance.

Finally, we compare our results against previous works. Our approach outperforms the results obtained by unsupervised systems on the noun portion of the SemEval’07 dataset (Patwardhan et al., 2007; Mohammad et al., 2007), achieving recall of .507 vs. .497 (a higher result obtained by Mohammad et al.). Table 4 presents LCS results separately for each of the source domains of the Koeling dataset for comparison purposes. Previous results using PPR and uniform context weighting reported by Agirre et al. (2009) (AL&S’09) are substantially lower than our baseline; we mainly attribute this to the different version of WordNet used. Huang and Lu (2011) proposed a manually-tuned syntax-based context selection and weighting formula. They applied it in combination with the GV method, reporting improvement on BNC sentences only. Our baseline result using GV was lower (.382 vs. .390), however LCS yielded better final performance (.401 vs. .397). Compared with their work, we use learning and model richer types of evidence; with PPR and LCS, we report best results on the BNC sentences. Table 4 details also recent results obtained for the BNC and Sports portions of the dataset. Ponzetto andNavigli (2010) (P&N’10) enriched the WordNet graph with additional relations projected onto the graph from Wikipedia; the table reports their best results using a graph centrality measure (Navigli and Lapata, 2010). Raviv and Markovitch (2012) (R&M’12) reported state-of-the-art performance in the specialized domains using Wikipedia as the reference knowledge base. Each individual context word is represented in their work as a weighted vector of Wikipedia concepts, where sense inference is performed by maximizing cosine similarity between the centroid of the context vectors and a vector representation of each word sense. Our results using PPR and LCS exceed or roughly match their results without using the Wikipedia resource.

4 Conclusion

We presented a learning framework that identifies useful contextual cues for knowledge-based sense disambiguation. The generated models are non-lexicalized, and are therefore applicable to new word types. Existing approaches pay little attention to context selection, or perform simplistic context modeling, whereas the proposed approach effectively consolidates diverse types of evidence. In the future, we are interested in representing additional word relatedness measures in this framework, such as embedding-based word similarity (Wang et al., 2015). We are further interested in creating specialized models that fit different word classes, e.g., of particular part-of-speech. In general, the proposed approach may prove beneficial for additional tasks that model word meaning in context, such as lexical substitution and sense induction.

Acknowledgments

We wish to thank Ido Dagan, Shuly Wintner and the anonymous reviewers for their useful comments. This work was supported by BSF under grant 2010090.

References


Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2009. Knowledge-based WSD on specific domains: performing better than generic supervised WSD. In Proceedings of IJCAI.


Lushan Han, Tim Finin, Paul McNamee, Anupam Joshi, and Yelema Yesha. 2013. Improving word similarity by augmenting PMI with estimates of word polysemy. IEEE Transactions on Knowledge and Data Engineering (TKDE), 25(6).

3They used WordNet 1.7, while we use version 3.0. Large performance gaps due to different versions of WordNet were reported elsewhere (Agirre and Soroa, 2009).


Thad Hughes and Daniel Ramage. 2007. Lexical semantic relatedness with random graph walks. In Proceedings of EMNLP.


Peter Turney. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of ECML.
