Memory-Based Word Sense Disambiguation

Jorn Veenstra, Antal van den Bosch, Sabine Buchholz, Walter Daelemans, Jakub Zavrel

ILK, Tilburg University, PO Box 90153, 5000 LE Tilburg, the Netherlands

Abstract. We describe a memory-based classification approach for word sense disambiguation. In this approach, a semantic word expert is automatically trained using memory-based learning. The training data consists of POS-tagged corpus examples and selected information from dictionary entries. The general approach is completely automatic; it only relies on the availability of a relatively small number of annotated examples for each sense of each word to be disambiguated, and not on human linguistic or lexicographic judgments. It is therefore easily adaptable and portable.

1. Introduction

For training the experts, and to speed up our approach, we do not employ standard machine learning algorithms. Instead, we perform pre-processing of information sources, (i) the POS-tagged corpus examples, (ii) the selected information from dictionary entries. The general approach is completely automatic; it only relies on the availability of a relatively small number of annotated examples for each sense of each word to be disambiguated, and not on human linguistic or lexicographic judgments. It is therefore easily adaptable and portable.

In this paper we describe a memory-based approach to learning word sense disambiguation.
In the remainder of this paper, we describe the different memory-based learning algorithms used, discuss the setup of our memory-based classification architecture for WSD, and report the generalization accuracy on the senseval data both for cross-validation on the training data, and for the final run on the evaluation data.

2. Memory-Based Learning

Memory-Based Learning (MBL) keeps all training data in memory and abstracts at classification time by extrapolating a class from the most similar item(s) in memory (i.e., it is a lazy learning method instead of the more common eager learning approaches). Eager learning methods "forget" information, because of their pruning and frequency-based abstraction methods. Moreover, the automatic feature weighting in the similarity metric of a memory-based learner makes the approach well-suited for domains with large numbers of features from heterogeneous sources, as it embodies a smoothing-by-similarity method when data is sparse (Zavrel and Daelemans, 1997).

For our experiments, we have used TiMBL, an MBL software package developed in our group (Daelemans et al., 1998). TiMBL includes the following variants of MBL:

- **ib**: The distance between a test item and each memory item is defined as the number of features for which they have a different value. In the TiMBL interface, a test item and each memory item is defined as the number of features for which they have a different value.

- **ib-ig**: In most cases, not all features are equally relevant for solving the task; this variant uses information gain (an information-theoretic notion measuring the reduction of uncertainty about the class to be predicted when knowing the value of a feature) to weight the cost of a feature value mismatch during comparison.

- **ib-mvdm**: For typical symbolic (nominal) features, values are not ordered. In the previous variants, mismatches between values are all interpreted as equally important, regardless of how similar (in terms of classification behaviour) the values are. We adopted the modified value difference metric to assign a different distance between each pair of values.

- **ib-mvdm-ig**: mvdm with ig weighting.

- **IGTree**: In this variant, an oblivious decision tree is created with features as tests, and ordered according to information gain of features.

TiMBL is available from: http://ilk.kub.nl/chum.tex; see the TiMBL homepage for more information.

In the remainder of this paper, we describe the different memory-based learning algorithms used, discuss the setup of our memory-based classification architecture for WSD, and report the generalization accuracy on the senseval data both for cross-validation on the training data, and for the final run on the evaluation data.
3. System Architecture and Experiments

For more references and information about these algorithms we refer to (Daelemans et al., 1998; Daelemans et al., 1999).
Although we did strip the letter suffixes (such as -s, -x) except for the -p suffix.

The following three properties (i) the word occurs in more than one
and the other (ii) if it is not a word is a keyword for a sense it drops
number of keywords per sense; these keywords are then used as bindings

effectively the sense in the work of Ng and Lo (1996), and extracts a

Although we will call these words the keywords, the method is
dictionary, the we will call these words the keywords. The method is
enough of the corpus the and if (the ex., the ex. and sentences in the
identified and only a portion of the dictionary's words are selected,
features used only a portion of the dictionary's words to extract the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the

The dictionary contains a number of fields for each sense, these underwrite the
Senses: The sense dictionary and a comparison to other systems participated in the shared task, and have standard metrics. For an overview of the shared task, the sense dictionary can be found in the shared task manual. The three metrics used are given in the shared task manual. The three metrics used are given in the shared task manual.

Definition Features
In addition to the keywords that passed the above selection, we use all open class words (nouns, adjectives, verbs, and adverbs) in the definition field. The definition field and the dictionary are used as features. The dictionary is only used for this purpose, and is not converted to a training case.

Post-processing
The definition field and the dictionary are used as features. The definition field and the dictionary are used as features. The dictionary is only used for this purpose, and is not converted to a training case.

Results
In this section we present the results we obtained with the optimal choice of metrics and feature construction parameters found with 10-fold cross validation on the training data, and the results on the evaluation data, as measured by the senseval coordination team.

For comparison we also provide the baseline results (on the training data), obtained by always choosing the most frequent sense. Table I shows the results per word. The algorithm and metric applied are indicated in the metric column; the value of k in the third column; the values of M_1, M_2, and M_3 in the next column; the accuracy with the optimal settings can be found in the 'train/opt' column; and the accuracy obtained with the default setting (M_1 = 0, M_2 = 5, M_3 = 5; the default suggested by Ng & Lee) and algorithm (ib/mvd, k = 1, no weighting) is given in the column 'train/def'. The three rightmost columns give the scores on the evaluation data, measured by the fine-grained, medium, and coarse standard respectively. For an overview of the scoring policy and a comparison to other systems participating in the shared task, the sense dictionary and a comparison to other systems participated in the shared task, and have standard metrics. For an overview of the shared task, the sense dictionary can be found in the shared task manual.
Conclusion

A memory-based architecture for word sense disambiguation does not require any hand-crafted linguistic knowledge, but only annotated training examples. Since for the present sense task dictionary information was available, we made use of this well, and it was easily accommodated in the learning algorithm.

We believe that ML is worth studying in domains such as WSD, where large numbers of features and sparseness of data interact to make life difficult for many other (e.g., probabilistic) machine learning methods, and where nonetheless every infrequent or exceptional information may prove to be essential for good performance.

References


Daelemans, W., A. Van den Bosch, and J. Zavrel: 1999, `Forgetting exceptions is harmful in language learning'. Machine Learning, Special issue on Natural Language Learning.


Acknowledgements


Conclusion

A memory-based architecture for word sense disambiguation does not require any hand-crafted linguistic knowledge, but only annotated training examples. Since for the present sense task dictionary information was available, we made use of this well, and it was easily accommodated in the learning algorithm.

We believe that ML is worth studying in domains such as WSD, where large numbers of features and sparseness of data interact to make life difficult for many other (e.g., probabilistic) machine learning methods, and where nonetheless every infrequent or exceptional information may prove to be essential for good performance.

References


Daelemans, W., A. Van den Bosch, and J. Zavrel: 1999, `Forgetting exceptions is harmful in language learning'. Machine Learning, Special issue on Natural Language Learning.


Acknowledgements

Acknowledgements

This research was done in the context of the "Induction of Linguistic Knowledge" (ILK) research programme, which is funded by the Netherlands Foundation for Scientific Research (NWO).
<table>
<thead>
<tr>
<th>Layer</th>
<th>Metric</th>
<th>K</th>
<th>N-1</th>
<th>N-2</th>
<th>N-3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
</tbody>
</table>

The scores on the evaluation set were computed by the word coordinators. The scores are average scores over the percentages in this table. The scores are based on the detailed and optional sets of the training set (average of 100 predictions).