Arabic/English Word Translation Disambiguation using Parallel Corpora and Matching Schemes

Farag Ahmed and Andreas Nürnberger

Data and Knowledge Engineering Group
Faculty of Computer Science
Otto-von-Guericke-University of Magdeburg
Building 29, Universitätsplatz 2
39106 Magdeburg
{farag.ahmed, andreas.nuernberger}@ovgu.de

Abstract. The limited coverage of available Arabic language lexicons causes a serious challenge in Arabic cross language information retrieval. Translation in cross language information retrieval consists of assigning one of the semantic representation terms in the target language to the intended query. Despite the problem of the completeness of the dictionary, we also face the problem of which one of the translations proposed by the dictionary for each query term should be included in the query translations. In this paper, we describe the implementation and evaluation of an Arabic/English word translation disambiguation approach that is based on exploiting a large bilingual corpus and statistical co-occurrence to find the correct sense for the query translations terms. The correct word translations of the given query term are determined based on their cohesion with words in the training corpus and a special similarity score measure. The specific properties of the Arabic language that frequently hinder the correct match are taken into account.

Keywords: Word Translation Disambiguation (WTD), Arabic language, Parallel Corpus, Naïve Bayesian Classifiers

1 Introduction

The meaning of a word may vary significantly according to the context in which it occurs. As a result, it is possible that some words can have multiple meanings. This problem is even more complicated when those words are translated from one language into others. Therefore there is a need to disambiguate the ambiguous words that occur during the translations. The word translation disambiguation, in general, is the process of determining the right sense of an ambiguous word given the context in which the ambiguous word occurs. We can define this word sense disambiguation (WSD) problem, as the association of an occurrence of an ambiguous word with one of its proper sense.

Diacritization in Arabic (sometimes called vocalization or voweling) can be defined as a symbol over and underscored letters, which are used to indicate the proper pronunciations as well as for disambiguation purposes. The absence of diacritization in Arabic texts poses a real challenge for Arabic natural language

processing as well as for translation, leading to high ambiguity the absence of the diacritization in most of the Arabic printed media or on the Internet web sites lead to high ambiguity. Thus, the probability that a single word can have multiple meanings is even a lot higher. For example, the Arabic word "يعد" can have these translations in English (*Promise, Prepare, count, return, bring back*) or the Arabic word "علم" can have these possible translations (*flag, science, he knew, it was known, he taught, he was taught*). The task of disambiguation therefore involves two processes: Firstly, identifying all senses for every word relevant, secondly assign each time this word occurs the appropriate sense to it. For the first step, this can be done using a list of senses for each of the ambiguous words existing in everyday dictionaries. The second step can be done – while still some uncertainty remains – by analyzing of the context in which the ambiguous word occurs, or by use of an external knowledge source, such as lexical resources. It is very important to consider the source of the disambiguation information, the way of constructing the rules using this information and the criteria of selecting the proper sense for the ambiguous word, using these rules. From the machine learning point of view approaches for WSD can be classified into three categories: supervised learning, unsupervised learning, and combinations of them.

2 Word Sense Disambiguation Approaches

Several methods for word sense disambiguation using a supervised learning technique have been proposed. For example, Naïve Bayesian [1], Decision List [2], Nearest Neighbor [3], Transformation Based Learning [4], Winnow [5], Boosting [6], and Naïve Bayesian Ensemble [7]. For all of these approaches, the one using Naïve Bayesian Ensemble is reported as the best performance for word sense disambiguation tasks with respect to data set used [7]. The idea behind all approaches listed above is that it is nearly always possible to determine the sense of the ambiguous word by considering its context, and thus all methods attempt to build a classifier, using features that represent the context of the ambiguous word. In addition to supervised approaches, unsupervised approaches and combinations of them have been also proposed for the same purpose. For example, [8] proposed an Automatic word sense discrimination which divides the occurrences of a word into a number of classes by determining for any two occurrences whether they belong to the same sense or not, which is then used for the full word sense disambiguation task. Examples of unsupervised approaches were proposed by [9][10]. [11] Proposed an unsupervised learning method using the Expectation-Maximization (EM) algorithm for text classification problems, which then was improved by [12] in order to apply it to the WSD problem. [13] Combined both supervised and unsupervised lexical knowledge methods for word sense disambiguation. [14] and [15] used rule-learning and neural networks respectively.

Corpora based methods for word sense disambiguation have also been studied. Corpora based methods provide an alternative solution for overcoming the lexical acquisition bottleneck, by gathering information directly from textual data. In the last few years, the natural data in electronic form has been increased, which helps the WSD researches to extend the coverage of the existing system or train a new system.

For example, [16] used the parallel, aligned Hansard Corpus of Canadian Parliamentary debates for WSD, [17] using monolingual corpora of Hebrew and German.

3 The Proposed Approach

Our approach is based on the idea of the Naïve Bayesian Algorithm [18]. We exploit the distribution of words and related words in parallel corpora, taking into account that the morphological inflection differs across the source and target languages.

The Naïve Bayesian Algorithm was first used for general classification problems. For WSD problems it had been used for the first time in [16]. It is based on the assumption that all features representing the problem are conditionally independent given the value of classification variables. For a word sense disambiguation tasks, giving a word *W*, candidate classification variables $S = (s_1, s_2, ..., s_n)$ that represent the senses of the ambiguous word, and the features $F = (f_1, f_2, ..., f_m)$ that describe the context in which an ambiguous word occurs, the Naïve Bayesian finds the proper sense s_i for the ambiguous word W by selecting the sense that maximizes the conditional probability given *F* and *S*. The Naive Bayesian probability estimate for a sense s_i can be defined as follows:

$$P(s_i \mid f_1, f_2, ..., f_n) = p(s_i) \prod_{j=0}^{m} p(f_j \mid s_i) \qquad (1)$$

The sense s_i of a polysemous word w_{amb} in the source language is defined by a synonym set (one or more of its translations) in the target language. The features for WSD, which is useful for identifying the correct sense of the ambiguous words, can be terms such as words or collocations of words. Features are extracted from the parallel corpus in the context of the ambiguous word. The conditional probabilities of the features $F = (f_1, f_2, ..., f_m)$ with observation of sense s_i, $P(f_j \mid s_i)$ and the probability of sense s_i, $P(s_i)$ are computed using maximum-likelihood estimates with $P(f_j \mid s_i) = C(f_j, s_i) / C(s_i)$ and $P(s_i) = C(s_i) / N$. $C(f_j, s_i)$ denotes the number of times feature f_j and sense s_i have been seen together in the training set. $C(s_i)$ denotes the number of occurrences of s_i in the training set and N is the total number of occurrences of the ambiguous word w_{amb} in the training dataset.

3.1 Feature Selection

The selection of an effective representation of the context (features) plays an essential role in WSD. The proposed approach is based on building different classifiers from different subset of features and combinations of them. Those features are obtained from the user query terms (not counting the ambiguous terms), topic context and word inflectional form in the topic context and combinations of them.

In our algorithm, query terms are represented as sets of features on which the learning algorithm is trained. Topic context is represented by a bag of surrounding words in a large context of the ambiguous word:

$$F = \{w_{w_{amb-k}}, \ldots, w_{w_{amb-2}}, w_{w_{amb-1}}, w_{amb}, w_{w_{amb+1}}, w_{w_{amb+2}}, \ldots, w_{w_{amb+k}}, q_1, q_2, \ldots, q_n\},$$

where k is the context size, w_{amb} is the ambigious word and *amb* its position. The ambigious word and the words in the the context can be replaced by their in-flectional forms. These forms and their context can be used as additional features. Thus, we obtain *F'* which contains in addition to the ambigious word w_{amb} and its context the inflectional forms $w_{\text{inf}.}$ of the given sense and their context:

$$F' = F \bigcup_{i=0}^{l} \{w_{w_{\text{inf}_i-k}}, \ldots, w_{w_{\text{inf}_i-2}}, w_{w_{\text{inf}_i-1}}, w_{\text{inf}_i}, w_{w_{\text{inf}_i+1}}, w_{w_{\text{inf}_i+2}}, \ldots, w_{w_{\text{inf}_i+k}}\}.$$

In addition, we count for each context word the number of occurrences of this word and all its inflectional forms.

3.2 General Overview of the System

As Figure 1 shows, the system starts to process the user query. The input is a natural language query Q. The query is then parsed into several words $q_1, q_2, q_3, .., q_n$. Each word is then further processed independent of the other words. Since the dictionary does not consist of all word forms of the translated word, instead only the root form, for each q_m in our query, we find its morphological root by using the araMorph tool[1].

After finding the morphological root of each term in the query, the query term will be translated. In case the query term has more than one translation, the model will provide a list of translations (sense inventory) for each of the ambiguous query terms. Based on the obtained sense inventory for the ambiguous query term, the disambiguation process can be initiated. The algorithm starts by computing the scores of the individual synonym sets. This is done by exploiting the parallel corpora in which the Arabic version of the translated sentences matches words or fragments of the user query, while matched words of the query must map to at least two words that are nearby in the corpus sentence. These words could be represented in surface form or in one of its inflectional forms. Therefore, and to increase the matching score quality, special similarity score measures will be applied in order to detect all word form variants in the translation sentences in the training corpora. Since the Arabic version of the translation sentences in the bilingual corpora matches fragments in the user query, the score of the individual synonym sets can be computed based on the features that represent the context of the ambiguous word. As additional features the words in the topic context can be replaced by their inflectional form. Once we have determined the features, the score of each of the sense sets can be computed. The sense which matches the highest number of features will be considered as the correct sense of the ambiguous query term and then it will be the best sense that describes the meaning of the ambiguous query term in the context.

1 http://www.nongnu.org/aramorph/

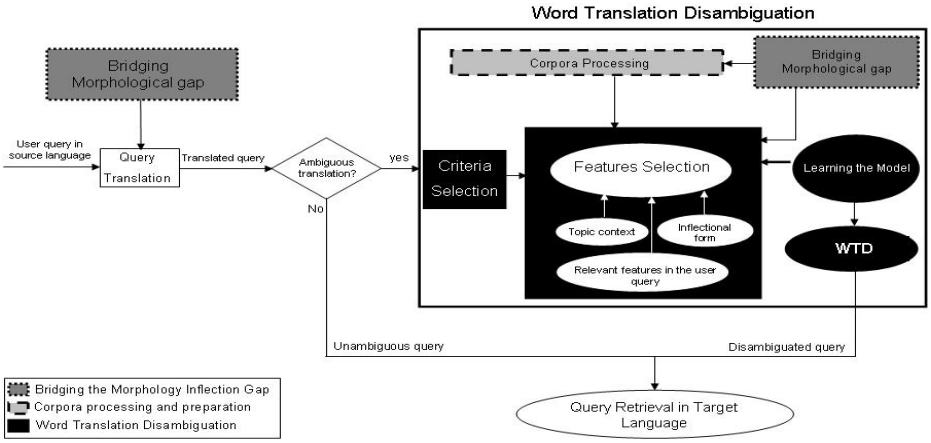

Fig. 1. General overview of the system

4 Evaluation

We evaluated our approach through an experiment using the Arabic/English parallel corpus aligned at sentence level. We selected 30 Arabic queries semi-randomly from the corpus as test set. We ensured that these queries contained at least one ambiguous word with multiple English translations. Furthermore, these ambiguous words had to have a higher frequencies compared with other words in the training data, ensuring that these words will appear in different situations in the training data. Furthermore, ambiguous words with high frequency sense were preferred. The sense (multiple translations) of the ambiguous words was obtained from the dictionary. The number of senses per test word ranged from 2 to 9, and the average was 4. For each test word, training data were required by the algorithm to select the proper sense. The algorithm was learned using more than 93,123 parallel sentences. The results of the algorithm were compared with the manually selected sense. For our evaluation we computed applicability and precision [29] based on different classifiers from different subsets of features and combinations of them as described in Sect. 3.1. The applicability is the proportion of the ambiguous words that the algorithm could disambiguate. The precision is the proportion of the corrected disambiguated senses for the ambiguous word. The overall results show that the performance varies according to the user query words. As expected, our approach is better in the case of long queries and worse in short query, especially the one consisting of only two words. The reason for the poor performance is that for queries consisting of only few words the features that are extracted from the query terms very often appear in context of different senses. Table 1 shows the overall performance of the algorithm based on building two classifiers from different subsets of features and combinations of them. As shown in Table 1, the performance of the algorithm is poor when using the basic word form. The reason for that is that the Arabic word can be represented not just in its basic form, but in many inflectional forms and so we will have more training sentences that will be visible to the algorithm for disambiguation.

Table 1. The overall performance using Applicability and Precision

classifiers	Applicability	Precision
Query term + Topic context (Basic form)	52 %	68 %
Query term+ Topic context (Basic & Inflectional form)	82 %	93 %

References

1. Gale, K. Church, and D. Yarowsky.: A Method for Disambiguating Word Senses in a Large Corpus. Computers and Humanities, vol. 26, pp. 415-439 (1992a).
2. Yarowsky.: Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, pp. 88-95 (1994).
3. T. Ng and H. B. Lee.: Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-based Approach. In Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, pp. 40-47 (1996)
4. Mangu and E. Brill.: Automatic rule acquisition for spelling correction. In Proceedings of the 14th International Conference on Machine Learning pp. 187-194.
5. R. Golding and D. Roth. A Winnow-Based Approach to Context-Sensitive Spelling Correction. Machine Learning, vol. 34, pp. 107-130 (1999)
6. Escudero, Gerard, Lluís Màrquez & German Rigau.: Boosting applied to word sense disambiguation. Proceedings of the 12th European Conference on Machine Learning (ECML), Barcelona, Spain, 129-141 (2000)
7. T. Pedersen.: A simple approach to building ensembles of Naive Bayesian classifiers for word sense disambiguation. In Proceedings of the 1st Annual Meeting of the North American Chapter of the Association for Computation Linguistics, pp. 63–69, Seattle, WA (2000)
8. Schütze, H.: Automatic WS discrimination. Computational Linguistics, 24(1):97-124 (1998)
9 .K. C. Litkowski.: Senseval: The cl research experience. In Computers and the Humanities, 34(1-2), pp. 153-158 (2000)
10. Dekang Lin.: Word sense disambiguation with a similarity based smoothed l brary. In Computers and the Humanities: Special Issue on Senseval, 34:147-152 (2000)
11. Nigam, McCallum, Thrun, and Tom Mitchell.: Text Classification from Labeled and Unlabeled Documents using EM. Machine Learning, 39(2/3):103–134 (2000)
12. Shinnou , Sasaki.: Unsupervised learning of word sense disambiguation rules by estimating an optimum iteration number in the EM algorithm, Proceedings of the 7th conference on Natural language learning at HLT-NAACL,p.41-48, Edmonton, Canada (2003)
13. E. Agirre, J Atserias, L.Padr, and G.Rigau.: Combining supervised and unsupervised lexical knowledge methods for word sense disambiguation. In Computers and the Humanities, Special Issue on SensEval. Eds. Martha Palmer &Adam Kilgarriff. 34:1,2 (2000)
14. David Yarowsky.: Unsupervised word sense disambiguation rivaling supervised methods. In Meeting of the Association for Computational Linguistics, pages 189.196 (1995)
15. Towell and E. Voothees.: Disambiguating Highly Ambiguous Words. Computational Linguistics, vol. 24, no. 1, pp. 125-146 (1998)
16. Gale, W. A., Church, K. W. & Yarowsky, D.: Using bilingual materials to develop word sense disambiguation methods. Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in MT (TMI'92), Montréal, 101-112 (. (1992))
17. Dagan, Ido and Itai, Alon.: Word sense disambiguation using a second language monolingual corpus. Computational Linguistics, 20(4):563-596 ((1994)
18. Duda, R. O. and Hart, P. E.: Pattern Classification and Scene Analysis, John Wiley, (1973)