Proceedings of the International Multiconference on
Computer Science and Information Technology, pp. 331 – 338

ISBN 978-83-60810-14-9
ISSN 1896-7094

Arabic/English Word Translation Disambiguation Approach based on Naive Bayesian Classifier

Farag Ahmed
Data and Knowledge Engineering Group
Faculty of Computer Science
Otto-von-Guericke-University of Magdeburg
39106 Magdeburg, Germany
Email: farag.ahmed@ovgu.de

Andreas Nürnberger
Data and Knowledge Engineering Group
Faculty of Computer Science
Otto-von-Guericke-University of Magdeburg
39106 Magdeburg, Germany
Email: andreas.nuernberger@ovgu.de

***Abstract*—We present a word sense disambiguation approach with application in machine translation from Arabic to English. The approach consists of two main steps: First, a natural language processing method that deals with the rich morphology of Arabic language and second, the translation including word sense disambiguation. The main innovative features of this approach are the adaptation of the Naïve Bayesian approach with new features to consider the Arabic language properties and the exploitation of a large parallel corpus to find the correct sense based on its cohesion with words in the training corpus. We expect that the resulting system will overcome the problem of the absence of the vowel signs, which is the main reason for the translation ambiguity between Arabic and other languages.**

I. Introduction

INITIALLY, online documents were used predominately by English speakers. Nowadays more than half (50.4%)[1] of web users speak a native language other than English. Therefore, it has become more important that documents of different languages and cultures are retrieved by web search engines in response to the user's request. Cross Language Information Retrieval CLIR allows the user to submit the query in one language and retrieve the results in different languages, providing an important capability that can help to meet that challenge. Cross-Language In-formation Retrieval (CLIR) approaches are typically divided into two main categories: approaches that exploit explicit representations of translation knowledge such as bilingual dictionaries or machine translation (MT) and approaches that extract useful translation knowledge from comparable or parallel corpora.

In the last few years, Arabic has become the major focus of many machine translation projects. Many rich resources are now available for Arabic. For example a GigaWord Arabic corpora, Arabic/English Parallel corpus, which contains several thousands sentence pairs of bilingual text for Arabic and English. The existence of these resources was a crucial factor in building effective translation tools. Bilingual dictionaries (Arabic with other languages) have been used in several Arabic CLIR experiments. However, bilingual dictionaries sometimes provide multiple translations for the same word, which need to be disambiguated. This is due to the fact, that the dictionary may have poor coverage; and it is difficult to select the correct sense of the translated word among all the translations provided by the dictionary.

This paper proposes a method to disambiguate the user translated query in order to determine the correct word translations of the given query terms by exploiting a large bilingual corpus and statistical co-occurrence. The Arabic language properties that hinder the correct match are taken into account by bridging the inflectional morphology gap for Arabic. We use one of the well-known Arabic morphological Analyzers [1] that includes the araMorph package, which we use to translate the user query from Arabic to the English language in order to obtain the sense inventory for each of the ambiguous user query terms.

A. Arabic language

Arabic is a Semitic language, consisting of 28 letters, and its basic feature is that most of its words are built up from, and can be analyzed down to common roots. The exceptions to this rule are common nouns and particles. Arabic is a highly inflectional language with 85% of words derived from tri-lateral roots. Nouns and verbs are derived from a closed set of around 10,000 roots [4]. Arabic has three genders, feminine, masculine, and neuter; and three numbers, singular, dual (represents 2 things), and plural. The specific characteristics of Arabic morphology make Arabic language particularly difficult for developing natural language processing methods for information retrieval. One of the main problems in retrieving Arabic language text is the variation in word forms, for example the Arabic word "kateb" (author) is built up from the root "ktb" (write). Prefixes and suffixes can be added to the words that have been built up from roots to add number or gender, for example adding the Arabic suffix "ان " (an) to the word "kateb" (author) will lead to the word "kateban" (authors) which represent dual masculine. What makes Arabic complicated to process is that Arabic nouns and verbs are heavily prefixed. The definite article "ال " (al) is always attached to nouns, and many conjunctions and prepositions are also attached as prefixes to nouns and verbs, hindering the retrieval of morphological variants of words [5]. Arabic is different from English and other Indo-European languages with respect to a number of important aspects. Words are written from right to left. It is mainly a consonantal language in its written forms, i.e. it excludes vowels. Its two main parts of speech are the verb and the noun in that word

[1] http://www.worldlingo.com/en/resources/languag_statistics.html

order and these consist, for the main part, of trilateral roots (three consonants forming the basis of noun forms that are derived from them). It is a morphologically complex language in that it provides flexibility in word formation: as briefly mentioned above, complex rules govern the creation of morphological variations, making it possible to form hundreds of words from one root [6].

Arabic poses a real translation challenge for many reasons; Arabic sentences are usually long and punctuation has no or little affect on interpretation of the text. Contextual analysis is important in Arabic in order to understand the exact meaning of some words. Characters are sometimes stretched for justified text (word will be spread over a bigger space than other words), which hinders the exact match for the same word. In Arabic, synonyms are very common, for example, "year" has three synonyms in Arabic **سنة ، حول ، عام** and all are widely used in everyday communication. Despite the previous issues and the complexity of Arabic morphology, which impedes the matching of the Arabic word, another real issue for the Arabic language is the absence of diacritization (sometimes called vocalization or voweling). Diacritization 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. Though the use of diacritization is extremely important for readability and understanding, diacritization are very rarely used in real life situations. They don't appear in most printed media in Arabic regions nor on Arabic Internet web sites. They are visible in religious texts such as the Quran, which is fully diacritized in order to prevent misinterpretation. Furthermore, the diacritization are present in children's books in school for learning purposes. For native speakers, the absence of diacritization is not an issue. They can easily understand the exact meaning of the word from the context, but for inexperienced learners as well as in computer usage, the absence of the diacritization is a real issue. When the texts are unvocalized, it is possible that several words have the same form but different meaning.

B.Tim Buckwalter Arabic morphological analyzer (BAMA)

(BAMA) is the most well known tool for analyzing Arabic texts. It consists of a main database of word forms that interact with other concatenation databases. An Arabic word is considered a concatenation of three regions: a prefix region, a stem region and a suffix region. The prefix and suffix regions can be null. Prefix and suffix lexicon entries cover all possible concatenations of Arabic prefixes and suffixes, respectively. Every word form is entered separately. It takes the stem as the base form and also provides information on the root. (BAMA) morphology reconstructs vowel marks and provides an English glossary. It returns all possible compositions of stems and affixes for a word. (BAMA) groups together stems with a similar meaning and associates it with a lemma ID. The (BAMA) contains 38,600 lemmas. For our work, we use the araMorph package. araMorph is a sophisticated java based Buckwalter analyzer.

II.Word Sense Disambiguation

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 (word sense disambiguation; WSD). We can define the WSD problem, as the association of an occurrence of an ambiguous word with one of it is proper sense. As described in the first section, the absence of the diacritization in most of the Arabic printed media or on the Internet web sites lead to high ambiguity. This makes the probability that the single word can have multiple meaning a lot higher. For example, the Arabic word "**يعد**" can have these meanings in English (Promise, Prepare, count, return, bring back) or the Arabic word "**علم**" can have these possible meanings (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 the appropriate sense each time this word occurs. 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 by the analysis of the context in which the ambiguous word occurs, or by the use of an external knowledge source, such as lexical resources as well as a hand-devised source, which provides data useful to assigning the appropriate sense for the ambiguous word. In the WSD task, 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. WSD is considered an important research problem and is assumed to be helpful for many applications such as machine translation (MT) and information retrieval. Approaches for WSD can be classified into three categorizations: supervised learning, unsupervised learning, and combinations of them.

A. Word Sense Disambiguation Approaches

Several methods for word sense disambiguation using a supervised learning technique have been proposed. For example, Naïve Bayesian [7], Decision List [8], Nearest Neighbor [9], Transformation Based Learning [10], Winnow [11], Boosting [12], and Naïve Bayesian Ensemble [13]. Using bilingual corpora to disambiguate words is leveraged by [14]. 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 the data set used [13]. The idea behind the previous approaches 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 for word sense disambiguation, unsupervised approaches and combinations of them have been also proposed for the same purpose. For example, [15] proposed an Auto-

matic 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 in [16][17][18][19][20][21]. [22] an unsupervised learning method using the Expectation-Maximization (EM) algorithm for text classification problems, which then was improved in [23] in order to apply it to the WSD problem. In [24] the combination of both supervised and unsupervised lexical knowledge methods for word sense disambiguation have been studied. In [25] and [26] rule-learning and neural networks have been used respectively.

Corpora based methods for word sense disambiguation has also been studied. Corpora based methods provide an alternative solution for overcoming the lexical acquisition bottleneck, by gathering information directly from textual data. Due to the expense of manual acquisition of lexical and disambiguation information, where all necessary information for disambiguation have to be manually provided, supervised approaches suffer from major limitations in their reliance on pre-defined knowledge sources, which affects their inability to handle large vocabulary in a wide variety of contexts. 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, in [27] and [28] the usage of parallel, aligned Hansard Corpus of Canadian Parliamentary debates for WSD has been performed, in [29] the authors use monolingual corpora of Hebrew and German for WSD. All of the previous studies were based on the assumption that the mapping between words and word senses is widely different from one language to another. Unlike machine translation and dictionaries, parallel corpora provide very high quality translation equivalents that have been produced by experienced translators, who associate the proper sense of a word based on the context that the ambiguous word is used in.

In the next section, we describe the proposed algorithm based on Naïve Bayesian classification, explaining the way of solving or at least relaxing the Arabic morphological issues. Afterward, we explain the features used to represent the context in which ambiguous words occur, followed by experimental results, which show the results of disambiguating some ambiguous words using a parallel corpus. This paper closes with a conclusion and future work.

III. Proposed Approach

Our approach is based on exploiting parallel texts in order to find the correct sense for the translated user query term. The minimum query length that the proposed approach accepts is two. Given the user query, the system begins by translating the query terms using the araMorph package. In case the system suggests more than one translation (senses inventory) for each of the query terms, the system then starts the disambiguation process to select the correct sense for the translated query terms. The disambiguation process starts by exploiting the parallel corpus, in which the Arabic version of the translation sentences matches fragments in the user query. A matched fragment must contain at least one word in the user query beside the ambiguous one. The words could be represented in surface form or in one of its variant 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 corpus.

A. Bridging the Inflectional morphology gap

The rich inflectional morphology languages face a challenge for machine translation systems. As it is not possible to include all word form variants in the dictionaries, inflected forms of words for those languages contain information that is not relevant for translation. The inflectional morphology differences between high inflectional language and poor inflectional language, presents a number of issues for the translation system as well as to disambiguation algorithms. This inflection gap causes a matching challenge when translating between rich inflectional morphology and relatively poor inflectional morphology language. It is possible to have the word in one form in the source language, while having the same word in few forms in the target language. This causes several issues for word translation disambiguation, where more unknown words forms will exist in the training data and will not be recognized as a relevant to the user query terms. As a result, it is possible to have lower matching scores for those words, even though there is a high occurrence of them in the training data.

The aim of this initial step is to alleviate the Arabic language morphology issues, which has to be done before accessing the Arabic language by the disambiguation algorithm. In order to deal with Arabic morphology issues, we used araMorph package [1].

To describe the problem more clearly, we consider for simplicity the Arabic word "دين" as described in section II. The absence of the diacritization from the Arabic printed media or the Internet web sites causes high ambiguity. The Arabic word "دين" has two translations in English (Religion or debt). We calculate the occurrences of this word in the training corpus for both senses. As it is shown in Table I the word "دين " was found in basic form for the sense (Religion) 49 times and for the sense (Debt) only 10 times.

As Table II shows, when we consider the inflectional form for the word "دين " we see that the occurrence of the inflectional form for the word "دين " with the sense (Religion) is 1192 and with the sense (Debt) is 231.

Table III shows sentence examples from the training corpus where the ambiguous word "دين "appears in basic or inflectional form with both senses. Detecting all word forms variants of the user query terms in the corpus is very essential when computing the score of the synonym sets, as it is shown in the Table II. More than 1386 sentences will be visible to the approach to disambiguate the ambiguous word "دين ". For more details about the word form variant detection and their impact on the retrieval performance, we refer the reader to our previous work [2][3].

TABLE I.
THE OCCURRENCE OF THE AMBIGUOUS WORD "دين "IN THE BASIC FORM FOR BOTH SENSES

The ambiguous word	senses	Occurrence in training data
		Basic form
دين	Religion	49
دين	Debt	10
Total		59

TABLE II.
THE OCCURRENCE OF THE INFLECTIONAL FORM FOR THE AMBIGUOUS WORD "دين "FOR BOTH SENSES

The ambiguous word	senses	Occurrence in training data
		Inflectional form
الدين	The Religion	75
والدين	And the Religion	22
الأديان	The Religions	45
والأديان	The Religions	7
الدينية	The Religious	63
والدينية	And the Religious	28
Total		240
الدين	The debt	860
والدين	And the debt	22
الديون	The debts.	255
والديون	And the debts.	9
Total		1146

In the following, our approach based on the Naïve Bayesian algorithm, where we learn words and their relationships from a parallel corpus, taking into account that the morphological inflection that differs across the source and target languages, is described.

B. Approach based on Naïve Bayesian Classifiers (NB)

The Naïve Bayesian Algorithm was first used for general classification problems. For WSD problems it had been used for the first time in [28]. The approach is based on the assumption that all features representing the problem are conditionally independent giving the value of classification variables. For a word sense disambiguation tasks, giving a word W, candidate classification variables $S = (s_1, s_2, ..., s_n)$, which represent the senses of the ambiguous word, and the feature $F = (f_1, f_1, ..., f_1)$ which 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 of occurring given the context. In other words, NB constructs rules that achieve high discrimination between occurrences of different word-senses by a probabilistic estimation. The Naive Bayesian estimation for the proper sense can be defined as follows:

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

TABLE III.
SENTENCES EXAMPLES FOR THE AMBIGUOUS WORD "دين " FOR BOTH SENSES IN BASIC AND INFLECTIONAL FORM

sense	Form	Arabic sentence	English translation
Religion	Basic	لأن الإسلام الذي هو *دين* حوار وانفتاح على الناس	because Islam, which is a ***religion*** of dialogue and openness to people
Debt	Basic	نضيف إلى ذلك أن الولايات المتحدة الأمريكية أكبر دولة مدينة في العالم، فلديها **400** مليار دولار عجزاً في ميزانيتها، يتم تمويلها عن طريق الاقتراض من المؤسسات الدولية والبنوك أو عن طريق تحويل هذا العجز إلى *دين* في الموازنة	In addition, the USA is the biggest debtorcountry in the world as it has a budget deficit of $400 billion which is financed through borrowing from international institutions and banks or through converting such a deficit into a budget ***debt***.
Religion	Infl.	ودعوا الوزير إلى التراجع عن قرار افتتاح المدرسة واستبدالها بمركز ثقافي ينشر تعاليم *الدين* والثقافة العربية	They called on the Minister to backtrack from that decision and to replace that school with a cultural centre promoting tenets of the ***religion*** and Arabic culture.
Debt	Infl.	وأكد الوزير أن الدين الخارجي على مصر هو في مستويات آمنة استناداً إلى ترتيبات جدولة *الدين* في نادي باريس	The minister emphasized that the foreign debt on Egypt was at safe levels due to the arrangements ***of debt*** scheduling in Paris Club.

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, that are 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, $P(f_j | s_i)$ and the probability of sense s_i, $P(s_i)$ are computed using maximum-likelihood estimates with $P(f_j | s_i) = C(f_j, s_i) / C(s_i)$ 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.

C. Features 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 ambiguous word and amb its position. The ambiguous word and the words in the the context can be replaced by their inflectional forms. These forms and their context can be used as additional features. Thus, we obtain F' which contains in addition to the ambiguous word w_{amb} and its context the inflectional forms $w_{\inf_i}$ of the given sense and their context, as it is shown in table II. Detecting all word form variants of the user query terms in the corpus will make 1386 sentences visible to the approach to disambiguate the ambiguous word "دين". In addition, we count for each context word the number of occurrences of this word and all its inflectional forms, i.e.

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

D. 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, only the root form, for each q_m in our query, we find its morphological root using the araMorph tool[2].

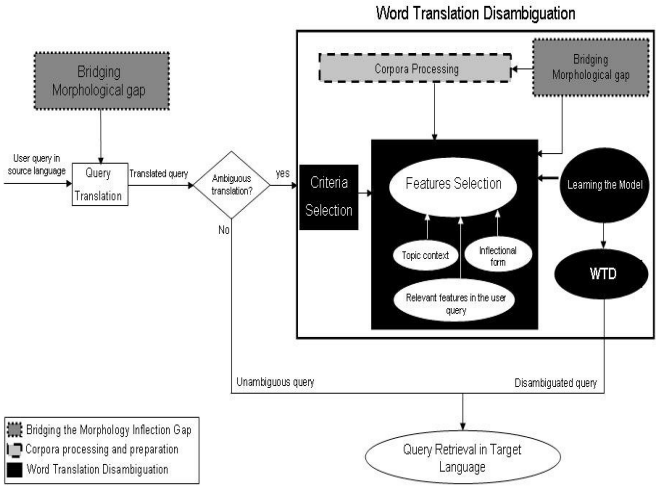

Fig. 1 General overview of the system

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.

E. Illustrative examples

To consider how the algorithm perform the disambiguation steps, consider the following simple query:

رسم جمركي للسلع

(A customs tax of commodities)

Step 1: The natural language query Q is parsed into several words $q_1, q_2, q_3, .., q_n$.

Step 2: For each q_m in the query, we find its morphological root.

Step 3: Translation of the query terms and creation of the sense inventory array in case of any for each of the query term is done. Table IV shows the sense inventory for each of the ambiguous query terms.

Step 4: The disambiguation process is initiated. The algorithm starts by computing the scores of the individual synonym sets:

- Number of times feature f_j and sense s_i have been seen together in the training set is computed.
- Number of occurrences of s_i in the training set is computed.
- The total number N of occurrences of the ambiguous word w_{amb} in the training dataset is computed.
- The disambiguation score is computed and the sense which matches the highest number of features will be considered as the correct sense of the ambiguous query term.

Table V shows the disambiguation scores of the individual synonym sets for each ambiguous query terms with other query terms. As Table V shows there are 135 possible translations set for the original query in source language.

[2] http://www.nongnu.org/aramorph/

TABLE IV.
SENSE INVENTORY FOR EACH OF THE AMBIGUOUS QUERY TERMS

Original Query term	Sense inventory (Possible English Translations)
رسم	[fee, tax, drawing, sketch, illustration, prescribe, trace, sketch, indicate, appoint]
جمركي	[customs, tariff, customs, control]
للسلع	[crack, rift, commodities, commercial, goods]

TABLE V.
DISAMBIGUATION SCORES FOR EACH POSSIBLE TRANSLATIONS SETS

S/N	query	score	S/N	query	score
1	fee AND (customs OR crack)	0	71	appoint AND (tariff OR crack)	0
2	fee AND (customs OR rift)	0	72	appoint AND (tariff OR rift)	0
3	fee AND (customs OR commodities)	0	73	appoint AND (tariff OR commodities)	0
4	fee AND (customs OR commercial)	0	74	appoint AND (tariff OR commercial)	0,00058
5	fee AND (customs OR goods)	0	75	appoint AND (tariff OR goods)	0
6	fee AND (control OR crack)	0	76	trace AND (customs OR crack)	0
7	fee AND (control OR rift)	0	77	trace AND (customs OR rift)	0
8	fee AND (control OR commodities)	0	78	trace AND (customs OR commodities)	0
9	fee AND (control OR commercial)	0	79	trace AND (customs OR commercial)	0
10	fee AND (control OR goods)	0	80	trace AND (customs OR goods)	0
11	fee AND (tariff OR crack)	0	81	trace AND (control OR crack)	0
12	fee AND (tariff OR rift)	0	82	trace AND (control OR rift)	0
13	fee AND (tariff OR commodities)	0	83	trace AND (control OR commodities)	0
14	fee AND (tariff OR commercial)	0	84	trace AND (control OR commercial)	0
15	fee AND (tariff OR goods)	0	85	trace AND (control OR goods)	0
16	tax AND (customs OR crack)	0,0484	86	trace AND (tariff OR crack)	0
17	tax AND (customs OR rift)	0,0484	87	trace AND (tariff OR rift)	0
18	**tax AND (customs OR commodities)**	**0,05948**	88	trace AND (tariff OR commodities)	0
19	tax AND (customs OR commercial)	0,05248	89	trace AND (tariff OR commercial)	0
20	tax AND (customs OR goods)	0,05539	90	trace AND (tariff OR goods)	0
21	tax AND (control OR crack)	0	91	sketch AND (customs OR crack)	0
22	tax AND (control OR rift)	0	92	sketch AND (customs OR rift)	0
23	tax AND (control OR commodities)	0,01224	93	sketch AND (customs OR commodities)	0
24	tax AND (control OR commercial)	0,00525	94	sketch AND (customs OR commercial)	0
25	tax AND (control OR goods)	0,01108	95	sketch AND (customs OR goods)	0
26	tax AND (tariff OR crack)	0,00175	96	sketch AND (control OR crack)	0
27	tax AND (tariff OR rift)	0,00175	97	sketch AND (control OR rift)	0
28	tax AND (tariff OR commodities)	0,01399	98	sketch AND (control OR commodities)	0
29	tax AND (tariff OR commercial)	0,007	99	sketch AND (control OR commercial)	0
30	tax AND (tariff OR goods)	0,01283	100	sketch AND (control OR goods)	0
31	prescribe AND (customs OR crack)	0	101	sketch AND (tariff OR crack)	0
32	prescribe AND (customs OR rift)	0	102	sketch AND (tariff OR rift)	0
33	prescribe AND (customs OR commodities)	0	103	sketch AND (tariff OR commodities)	0
34	prescribe AND (customs OR commercial)	0	104	sketch AND (tariff OR commercial)	0
35	prescribe AND (customs OR goods)	0	105	sketch AND (tariff OR goods)	0
36	prescribe AND (control OR crack)	0	106	drawing AND (customs OR crack)	0,00058
37	prescribe AND (control OR rift)	0	107	drawing AND (customs OR rift)	0,00058
38	prescribe AND (control OR commodities)	0	108	drawing AND (customs OR commodities)	0,00117
39	prescribe AND (control OR commercial)	0	109	drawing AND (customs OR commercial)	0,0035
40	prescribe AND (control OR goods)	0	110	drawing AND (customs OR goods)	0,00058
41	prescribe AND (tariff OR crack)	0	111	drawing AND (control OR crack)	0,00058
42	prescribe AND (tariff OR rift)	0	112	drawing AND (control OR rift)	0,00058
43	prescribe AND (tariff OR commodities)	0	113	drawing AND (control OR commodities)	0,00117
44	prescribe AND (tariff OR commercial)	0	114	drawing AND (control OR commercial)	0,0035
45	prescribe AND (tariff OR goods)	0	115	drawing AND (control OR goods)	0,00058
46	indicate AND (customs OR crack)	0	116	drawing AND (tariff OR crack)	0,00058
47	indicate AND (customs OR rift)	0	117	drawing AND (tariff OR rift)	0,00058
48	indicate AND (customs OR commodities)	0	118	drawing AND (tariff OR commodities)	0,00117
49	indicate AND (customs OR commercial)	0	119	drawing AND (tariff OR commercial)	0,0035
50	indicate AND (customs OR goods)	0,00117	120	drawing AND (tariff OR goods)	0,00058
51	indicate AND (control OR crack)	0,00058	121	illustration AND (customs OR crack)	0
52	indicate AND (control OR rift)	0,00058	122	illustration AND (customs OR rift)	0
53	indicate AND (control OR commodities)	0,00058	123	illustration AND (customs OR commodities)	0
54	indicate AND (control OR commercial)	0,00058	124	illustration AND (customs OR commercial)	0
55	indicate AND (control OR goods)	0,00175	125	illustration AND (customs OR goods)	0
56	indicate AND (tariff OR crack)	0	126	illustration AND (control OR crack)	0
57	indicate AND (tariff OR rift)	0	127	illustration AND (control OR rift)	0
58	indicate AND (tariff OR commodities)	0	128	illustration AND (control OR commodities)	0
59	indicate AND (tariff OR commercial)	0	129	illustration AND (control OR commercial)	0
60	indicate AND (tariff OR goods)	0,00117	130	illustration AND (control OR goods)	0
61	appoint AND (customs OR crack)	0	131	illustration AND (tariff OR crack)	0
62	appoint AND (customs OR rift)	0	132	illustration AND (tariff OR rift)	0
63	appoint AND (customs OR commodities)	0	133	illustration AND (tariff OR commodities)	0
64	appoint AND (customs OR commercial)	0,00058	134	illustration AND (tariff OR commercial)	0
65	appoint AND (customs OR goods)	0	135	illustration AND (tariff OR goods)	0
66	appoint AND (control OR crack)	0			
67	appoint AND (control OR rift)	0			
68	appoint AND (control OR commodities)	0			
69	appoint AND (control OR commercial)	0,00058			
70	appoint AND (control OR goods)	0			

F. Training data

The proposed algorithm was developed using Arabic/English parallel corpus[3]. This corpus contains Arabic news stories and their English translations. It was collected via Ummah Press Service from January 2001 to September 2004. It totals 8,439 story pairs (Documents), 68,685 sentence pairs, 93,120 segments pairs, 2 Million Arabic words and 2.5 Million English words. The corpus is aligned at sentence level.

IV. Evaluation

We evaluated our approach through an experiment using the Arabic/English parallel corpus aligned at sentence level. We selected 30 Arabic sentences from the corpus as queries to test the approach. These sentences have various lengths starting from two words up to five words, as future work the maximum query length will be extended. These queries had to contain at least one ambiguous word, which has multiple English translations. In order to enrich the evaluation set, these ambiguous words had to have higher frequencies compared with other words in the training data, ensuring that these words will appear in different contexts 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 two to nine, and the average was four. For each test word, training data were required by the algorithm to select the proper sense. The algorithm was applied to more than 93,123 parallel sentences. The results of the algorithm were compared with the manually selected sense.

For our evaluation, we built different classifiers from different subsets of features and combinations of them. The first classifier based on features that were obtained from the user query terms and topic context, which was represented by a bag of words in the context of the ambiguous word. The second classifier was based on the topic context and its inflectional form.

In order to evaluate the performance of the different classifiers, we used two measurements: applicability and precision [29]. 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 performance of our approach is summarized in Table IV. The sense, which is proposed by the algorithm was compared to the manually selected sense.

As it is expected the approach is better in the case of long query terms which provide more reach features and worse in short query, especially the one consisting of two words. We consider that the reason for the poor performance is that, when the query consists of few words it is possible that the features which are extracted from the query terms can appear in the context of different senses. For example, consider the query "**الدين الإسلامي**" (The Islamic religion). When the algorithm goes through the corpus, the ambiguous word "الدين" (The Religion or The debt) will be found in two different context whether in Religion or Debt context. The query term "**الإسلامي**" (Islamic) can be found in both contexts of the ambiguous word as (Islamic religion) or as a name of bank (Islamic Bank), which is the context of the second sense (Debt). One possible solution for this issue is query expansion. This can be done by exploiting the corpus and suggesting possible term expansion to the user. The user then confirms this term expansion, which will help to disambiguate the ambiguous query term when translating to the target language.

Another reason for the poor performance is that due to the morphological inflectional gap between languages such as Arabic the same word can be found in different forms. In order to increase the performance of the disambiguation process all of these forms need to be detected.

Table VI shows the overall performance of the algorithm based on building two classifiers from different subsets of features and combinations of them. Those features are user query terms, topic context and word inflectional form in topic context and combinations of them. As is shown in Table IV, the performance of the algorithm is poor when using the basic word form. The reason for 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 to disambiguate the ambiguous query terms.

Table VI.
The overall performance using Applicability and precision

classifiers	Applicability	Precision
Query term + Topic context	52 %	68 %
Query term+ feature Inflectional form	82 %	93 %

[3]http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004T18

References

[1] Tim Buckwalter, Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Catalog No.: LDC2002L49.

[2] Farag Ahmed and Andreas Nürnberger, N-Grams Conflation Approach for Arabic Text, In: Proceedings of the International Workshop on improving Non English Web Searching (iNEWS 07) In conjunction with the 30th Annual International (ACM SIGIR Conference). Amsterdam City, Netherlands, 2007, pp. 39-46.

[3] Farag Ahmed and Andreas Nürnberger, araSearch: Improving Arabic text retrieval via detection of word form variations, In: Proceedings of the 1st International Conference on Information Systems and Economic Intelligence (SIIE'2008) at Hammamet in Tunisia, 2008, pp. 309-323.

[4] Al-Fedaghi Sabah S. and Fawaz Al-Anzi, Anew algorithm to generate Arabic root-pattern forms. Proceedings of the 11th National Computer Conference, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia, 1989, pp. 04-07.

[5] Moukdad, H., Lost in Cyberspace: How do search engines handle Arabic queries? In Access to Information: Technologies, Skills, and Socio-Political Context. Proceedings of the 32nd Annual Conference of the Canadian Association for Information Science, Winnipeg, June 2004, pp. 3-5.

[6] Moukdad, H. and A. Large, Information retrieval from full-text Arabic databases: Can search engines designed for English do the job? Libri 51 (2), 2001, pp. 63-74.

[7] Gale, K. Church, and D. Yarowsky, A Method for Disambiguating Word Senses in a Large Corpus. Computers and Humanities, vol. 26, 1992a, pp. 415-439.

[8] 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, 1994, pp. 88-95.

[9] 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, 1996, pp. 40-47.
[10] Mangu and E. Brill, Automatic rule acquisition for spelling correction. In Proceedings of the 14th International Conference on Machine Learning, 1997, pp. 187-194.
[11] R. Golding and D. Roth, A Winnow-Based Approach to Context-Sensitive Spelling Correction. Machine Learning, vol. 34, 1999, pp. 107-130.
[12] Escudero, Gerard, Lluís Màrquez & German Rigau, Boosting applied to word sense disambiguation. Proceedings of the 12th European Conference on Machine Learning (ECML), Barcelona, Spain, 2000, pp. 129-141.
[13] T. Pedersen, A simple approach to building ensembles of Naive Bayesian classifiers for word sense disambiguation. In Proceedings of the First Annual Meeting of the North American Chapter of the Association for Computational Linguistics, Seattle, WA, May, 2000, pp. 63–69.
[14] Nancy Ide, N., Parallel translations as sense discriminators. SIGLEX99: Standardizing Lexical Resources, ACL99 Workshop, College Park, Maryland, 1999, pp. 52--61.
[15] Schütze, H.: Automatic word sense discrimination. Computational Linguistics, v.24, n.1, (1998) 97-124.
[16] K. C. Litkowski. 2000. Senseval: The cl research experience. In Computers and the Humanities, 34(1-2), pp. 153-158.
[17] Dekang Lin., Word sense disambiguation with a similarity based smoothed l brary. In Computers and the Humanities: Special Issue on Senseval, 2000, pp. 34:147-152.
[18] Philip Resnik., Selectional preference and sense disambiguation. In Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?, Washington, 1997, pp. 4-5.
[19] David Yarowsky, Word-sense disambiguation using statistical models of Ro-get's categories trained on large corpora. In Proceedings of COL-ING-92, Nantes, France, 1992, pp. 454.460.
[20] Indrajit Bhattacharya, Lise Getoor, Yoshua Bengio: Unsupervised Sense Disambiguation Using Bi-lingual Probabilistic Models. ACL 2004: 287-294.
[21] Hiroyuki Kaji, Yasutsugu Morimoto: Unsupervised Word-Sense Disambiguation Using Bilingual Comparable Corpora. IEICE Transactions 88-D(2), 2005, pp. 289-301.
[22] Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom Mitchell, Text Classification from Labeled and Unlabeled Documents using EM. Machine Learning, 39(2/3), 2000, pp. 103–134.
[23] Hiroyuki Shinnou , Minoru Sasaki, Unsupervised learning of word sense disambiguation rules by estimating an optimum iteration number in the EM algorithm, Proceedings of the seventh conference on Natural language learning at HLT-NAACL, Canada., May 31, 2003, Edmonton, pp. 41-48.
[24] 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 Double Issue on SensEval. Eds. Martha Palmer and Adam Kilgarriff, 2000, pp. 34:1,2.
[25] David Yarowsky, Unsupervised word sense disambiguation rivaling supervised methods. In Meeting of the Association for Computational Linguistics, 1995, pp. 189.196.
[26] Towell and E. Voothees, Disambiguating Highly Ambiguous Words. Computational Linguistics, vol. 24, no. 1, 1998, pp. 125-146.
[27] Brown, P. F., Lai, J. C. & Mercer, R. L., Aligning Sentences in Parallel Corpora, Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics. Berkeley, 1991, pp. 169-176.
[28] 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 Machine Translation (TMI'92), Montréal, 1992, pp. 101-112.
[29] Dagan, Ido and Itai, Alon, Word sense disambiguation using a second language monolingual corpus. Computational Linguistics, 20(4), 1994, pp. 563-596.
[30] Duda, R. O. and Hart, P. E.: Pattern Classification and Scene Analysis, John Wiley, 1973.