Automatic Approach for Word Sense Disambiguation Using Genetic Algorithms

Dr. Bushra Kh. AlSaidi
Computer Center
Collage of Economic and Administration/Baghdad University
Baghdad, Iraq

***Abstract*—Word sense disambiguation (WSD) is a significant field in computational linguistics as it is indispensable for many language understanding applications. Automatic processing of documents is made difficult because of the fact that many of the terms it contain ambiguous. Word Sense Disambiguation (WSD) systems try to solve these ambiguities and find the correct meaning. Genetic algorithms can be active to resolve this problem since they have been effectively applied for many optimization problems. In this paper, genetic algorithms proposed to solve the word sense disambiguation problem that can automatically select the intended meaning of a word in context without any additional resource. The proposed algorithm is evaluated on a collection of documents and produce's a lot of sense to the ambiguities word, the system creates dynamic, and up-todate word sense in a highly automatic method.**

Keywords—unsupervised method; genetic algorithms; word sense disambiguation; Natural Language Processing; Information Retrieval

I. INTRODUCTION

Most of the people use the web to find some contents. At searching process they never worry about ambiguities that occur between words .An ambiguous word is a word that has a lot of meaning in different contexts [2]. The context in which the ambiguous word appears is determined the sense of the word. When the person makes the search related to an ambiguous word, search engines shows all the results related to senses of the word. Several of them are relevant and others are irrelevant.

Word Sense Disambiguation (WSD) is the procedure of finding the senses of the words in textual context, when word has several meanings [5].WSD is a long-standing issue in Computational Linguistics, and has main effect in many real-world applications involve machine translation, information extraction, and information retrieval [6].Word Sense Disambiguation (WSD) systems try to solve these ambiguities and finding the accurate meaning.

There are two approaches of WSD [4]:

1) ***Deep Approaches:*** *This is built on world knowledge. But such knowledge is not existing in computers readable format except in some restricted domain so this approach is not very common. However if such knowledge exist than this approach will be much more precise than shallow approaches.*

Example: Man goes fishing for some bass.

Here knowledge is used to find the meaning of ‘bass’ in the sentence because an individual can go fishing for a kind of fish but not for low frequency sound. So here bass will refer to fish.

2) ***Shallow Approaches:*** *This approach does not depended on the world knowledge. An individual can understand the text over the surrounding words.*

Example: If ‘crane’ has words sky or fly near then it will refer to bird. If ‘crane’ has words parts or manufacture nearby it points to machine. The algorithm used in this paper considers as shallow approaches.

Dynamic languages grow by time such that even more work is required to develop new examples if new terms looked suddenly or gone. For instance, the word "rock" currently has the meaning of a stone as well as a music genre. To avoid preparing annotated corpora, effort needs to be oriented to new approaches in the knowledge-based unsupervised direction, one of the current trends to address WSD as a combinatorial optimization problem.[8].

The improvement of WSD systems is often expensive to acquiring the needed training data. The supervised methods are the best predictors of WSD difficulty, but the dependence on labeled training data limited them. The unsupervised approaches all implement well in many situations and can be applied widely. To avoid the problem of annotated corpora this paper presented an unsupervised approach based on genetic algorithms to solve the problem of ambiguity words that automatically find the sense of the word from the document's collection.

The rest of the paper is organized as follows: section II explains the related work. Section III introduces the proposed algorithm. In section IV the experimental part is presented. Section V explains the effect of some genetic algorithm factors in the performance of the proposed algorithm. Section VI explains the conclusions and section VII introduces the future work.

II. RELATED WORK

There exists important work on Word Sense Disambiguation for many languages using numerous approaches. The following paragraphs will explain some of the related work that used genetic algorithms to solve the problem of ambiguity words. Zhang, Zhou and Mmartin[3] proposed unsupervised genetic word sense disambiguation algorithm (GWSD). The algorithm using WordNet to describe possible senses for a set of words, to maximize the semantic similarity, genetic algorithm is applied on this set of words. Domain information and conceptual similarity function is used to calculate similarity between senses in WordNet. They proposed a weighted genetic word sense disambiguation algorithm (WGWSD) for word sense disambiguation in a general corpus. Experiments on SemCor are carried out to compare WGWSD with previous work. Azzini, Pereira, Dragony and Rettamanzi [1] proposed a supervised method to WSD based on neural networks shared with evolutionary algorithms. Big tagged datasets are considered for each sense of a polysemous word, and used to evolve an optimized neural network. kumara and singh[9] using genetic algorithm with Elitism for Hindi language. a lexical knowledge base and Wordnet for Hindi is used. The central focus is on word sense disambiguation using the context by applying GAs. Menai[6] proposed to use genetic and memetic algorithms to solve the word sense disambiguation problem , and apply them to Modern Standard Arabic. Its performance evaluated and compared against a naïve Bayes classifier. results show that genetic algorithms can reach more precise prediction than naïve Bayes classifier and memetic algorithms. Alsaeedan and Menai_ [11] proposed a self-adaptive GA for the WSD problem with an automatic modification of its mutation and crossover probabilities. The experimental results found on standard corpora (Senseval-2 (Task#1), SensEval-3 (Task#1), and SemEval-2007 (Task #7)) show that the presented algorithm considerably outperformed a genetic algorithm with standard genetic operators in recall and precision.

III. Word Sense Disambiguation Using Genetic Algorithms

Representation of the context in which an ambiguous word occurs has great effect to successfully applied machine learning methods for word sense disambiguation (WSD) problem [6].So to apply genetic algorithms to word sense disambiguation (WSD) the search space of the problem must be represented in suitable format. The proposed algorithm applied to WSD without any resource like dictionary or thesauri, it worked directly on the collection of the documents. To start the method, words that require to be disambiguated are compiled and kept in a text file. Each word is send to a search engine as a query to retrieve great number of Web pages. The Web pages are cleaned and control characters and HTML tags are removed. The Knowledge base contains a corpus of sentences consisting of the ambiguous words and a number of other words which co-occur with the ambiguous words. The proposed approach implemented in the following steps:

A. Preprocessing

The preprocessing procedure includes the following steps:

1) Tokenize each documents and performed word segmentation to get the word set.

2) Remove the stop word like (the ,an, and ….) from the word set.

3) Remove the ambiguity word from the word set

B. Applying genetic algorithm for WSD

After preprocessing step genetic algorithms applied as follows:

1) Population Representation and Initialization

GAs runs on a number of potential solutions, termed a population, containing many encoding of the parameter set simultaneously. To solve word sense disambiguation the population was initiated with 50 variables in length chromosome as following:

For each individual in the population do

A. *Pick a random number representing the length of the chromosome in the population.*

B. *Chose words randomly from the set of word resulted from the preprocessing step in number equal to the length of the individual.*

C. *Repeat steps a and b until the population size reached.*

After these three steps are completed the population initiated with 50 variables in length individuals.

The variable length individual was used because the sense of the word may have multiple words like the possible sense edge of river of the word bank.

```
For n=1 to population size do
    m=rand (1, maximum chromosome length)
    For j=1 to m do
        Chromosome [n][j]=word set[rand(1,word set size)]
    End
End
```

Fig. 1. Pesodocode of the population initialization

1) fitness function

The fitness function for each individual measures how many documents covered by these words sense represented in that individual in another word how much the sense generated by the algorithm represent the actual meaning of the word in that context. The fitness function doesn't reward sense that are more frequently used.

2) Termination of the GA

Since the algorithm is unsupervised, there are no a specific results that should be found by the algorithm therefore, the GA terminated after a prespecified number of generations and then all individuals in the population that have fitness function more than specified threshold represented the discovered senses of the ambiguous word.

The algorithm allows overlapping so one document may have more than one sense of the ambiguous word.

```
The Input : an ambiguous word and a collection of documents
The output : set of all possible senses of the ambiguous word
procedure GA
begin
t = 0;
initialize Pop(t);
evaluate Pop(t);
while not finished do
begin
t = t + 1;
 select Pop(t) from Pop(t-1);
reproduce pairs in Pop(t);
evaluate Pop(t);
end
end.
```

Fig. 2. The proposed algorithm

IV. Experiment

The method evaluated with some none ambiguous word like bank and as the Natural languages are dynamic in nature, with time new words emerge and usages of words tend to change. The method tested with the word python. Human beings can keep themselves updated with latest words or new vocabulary but to maintain same level of maintenance in updating latest knowledge for a machine needs continuous efforts, the genetic algorithm for WSD method is one of these effort. Table 1 below showed results of the system for the word ***bank*** ,the results show that the dominant sense of the word is a financial business building and there are many senses with different fitness value and other senses have little rate of appearance.

TABLE I. Systems Results for the word bank

Sense of word	Fitness function
Financial	70%
Building, business	60%
Services, group	30%
Financial ,Service	25%
Account ,money, security	18%
Fishmen	12%
Painting	10%
watershed	6%
sunset	4%
sand, wood	3%
right	3%
forest ,boat	1%
cutting	1%
autumn	1%
butterfly	1%

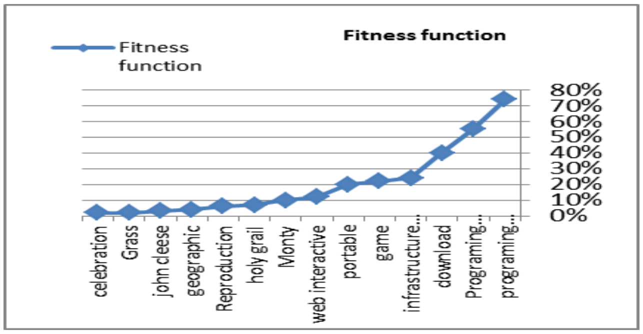

Fig. 3. The fitness of the senses discovered by the proposed algorithm for the word (bank)

Table 2 below showed results of the system for the word python, the results shows that the dominant sense of the word is programing language and the algorithm produces other senses with different values of fitness.

TABLE II. The Result of the System for the Word Python

Sense of word	Fitness function
programing language	74%
Programing tutorial	55%
download	40%
infrastructure platform	24%
game	29%
portable	20%
web interactive	12%
Monty	10%
holy grail	7%
Reproduction	6%
geographic	4%
john cleese	3%
Grass	2%
celebration	2%

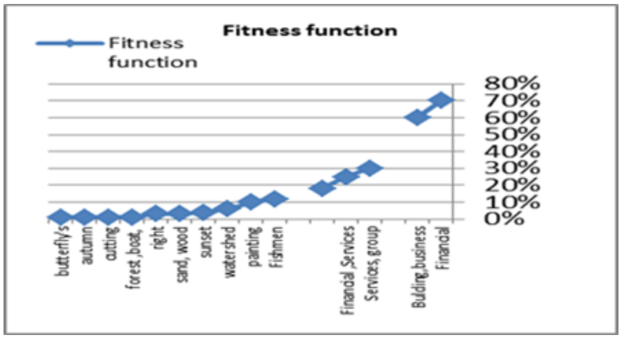

Fig. 4. The fitness of the senses discovered by the proposed algorithm for the word Python

V. Efect of Genetic Algorithms Paramters

In these experiments we are trying to study the effect of the genetic algorithm parameters in the performance of the proposed algorithm.

1) elitism

In these experiments we are trying to study the effect of the elitism operation in the performance of the proposed algorithm.

Genetic algorithm with the elitism operation saves the best population in the whole generation to be the solution and reject the new solutions that do not improve the existing ones.

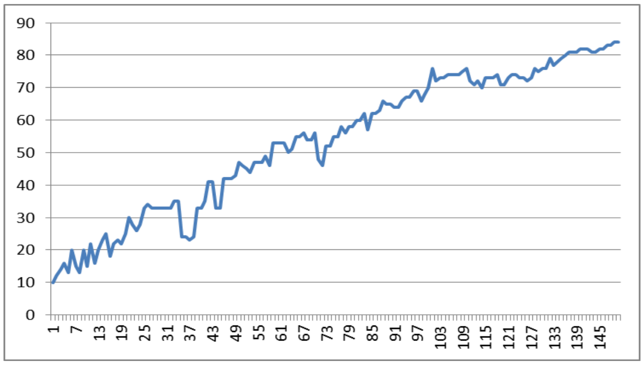

Fig. 5. Performance of proposed algorithm without elitism

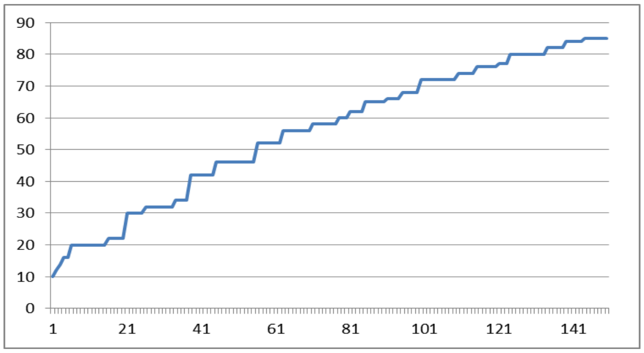

Fig. 6. Performance of proposed algorithm with elitism

VI. Conclusion

This paper proposed a word sense disambiguation method based on genetic algorithm. The results of the experiments reflect that the system works well in disambiguating the words without any need to an annotated example. The algorithm produced a wide range of word sense according to the context of the word in the document. Encoding the population with words taken directly from documents will be reduced the time required to reach the optimal solution ,this is considered as an employment of some prior knowledge and the results can be directly introduced to the user and don't need any post processing. The experiments on the dataset collocated from the web have shown that the algorithm achieves promising results in WSD field.

VII. Future Work

In the future it will be possible to study some genetic algorithm parameters like population size and maximum chromosome length**.** It is also possible application of other algorithms such as swarm intelligence algorithms and compares the results with the algorithm proposed in this research.

References

[1] A. Azzini, M. Dragoni, A. G. B. Tettamanzi "A lexicographic encoding for word sense disambiguation with evolutionary neural networks" ChapterAI*IA 2009: Emergent Perspectives in Artificial Intelligence Volume 5883 of the series Lecture Notes in Computer Science pp 192-201

[2] A. Di Marco, R. Navigli. "Clustering and diversifying web search results with graph-based word sense induction" computational linguistics 2013. Volume 39, number 3.

[3] C. Zhang , Y. Zhou ,T. Martin, "Genetic word sense disambiguation algorithm" IEEE, Intelligent Information Technology Application, 2008. IITA '08. Second International Symposium on (Volume:1) Page(s): 123-127

[4] E. Glenny, E. Agirre. "Word sense disambiguation: algorithms and applications" Springer, 2006, online, Available: books.google.co.in/books? isbn=1402048092.

[5] N. Fauceglia, Y.Lin, X. Ma, and E. Hovy. "Word sense disambiguation via PropStore and OntoNotes for event mention detection" Proceedings of the 3rd Workshop on EVENTS at the NAACL-HLT 2015, pages 11–15.

[6] M. Menai "Word sense disambiguation using an evolutionary approach" informatics 38(2014) 155-169

[7] P. Chen, C. Bowes, W. Ding, D. Brown. "A fully unsupervised word sense disambiguation method using dependency knowledge" Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pages 28–36, Boulder, Colorado, June 2009. c 2009 Association for Computational Linguistics.

[8] S. Abualhaija, K. zimmermann "D-Bees: Aa novel method inspired by bee colony optimization for solving word sense disambiguation" 2014 CoRR abs/1405.1406.

[9] S. Kumari1, P.Singh "Genetic algorithm based Hindi word disambigauation " Journal of Computer Science and Mobile Computing IJCSMC, Vol. 2, Issue. 5, May 2013, pg.139 – 1449

[10] T. H. Wang, J. Y. Rao, Q. Hu. "Supervised word sense disambiguation using semantic diffusion kernel", Engineering Applications of Artificial Intelligence, vol. 27, (2014), pp. 167-174.

[11] W. Alsaeedan, M. Menai "A Self-adaptive genetic algorithm for the word sense disambiguation problem" springer, Current Approaches in Applied Artificial Intelligence Volume 9101 of the series Lecture Notes in Computer Science pp 581-590 Date: 01 May 2015