International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 5 Issue: 6 635 – 639

Gujarati Word Sense Disambiguation using Genetic Algorithm

Zankhana B. Vaishnav
Assistant Professor, Dept.ofMaster of Computer Application
Sarvajanik College of Engineering & Technology
Surat, India
zankhana.vaishnav@scet.ac.in

Abstract— Genetic algorithms (GAs) have widely been investigated to solve hard optimization problems, including the word sense disambiguation (WSD). This problem asks to determine which sense of a polysemous word is used in a given context. Several approaches have been investigated for WSD in English, French, German and some Indo-Aryan languages like Hindi, Marathi, Malayalam, etc. however, research on WSD in Guajarati Language is relatively limited. In this paper, an approach for Guajarati WSD using Genetic algorithm has been proposed which uses Knowledge based approach where Indo-Aryan WordNet for Guajarati is used as lexical database for WSD.

Keywords-*Natural Language Processing, Word Sense Disambiguation, Genetic Algorithm, WordNet*

*****

I. INTRODUCTION

Natural language processing is related to human–computer interaction, where several challenges involve natural language understanding. Word sense disambiguation problem (WSD) consists in the computational assignment of a meaning to a word according to a particular context in which it occurs. A word can have number of senses, which is termed as ambiguity. This word sense disambiguation is an intermediate task, but rather is necessary at one level to accomplish most natural language processing tasks. For example, in following Gujarati sentences a word is having two different senses.

1. મોંઘવારીથીબધાવર્ગનાલોકોપરેશાનછે.
2. સાતનોવર્ગઓગણચાલીસથાયછે.

In first sentence, the word વર્ગ means class where as in second sentence, it means square of a number. There are many such words available in many languages which will carry different sense according to the context.

In specific terms, we can view a text T as a sequence of words ($w_1, w_2, \ldots, w_n$), and we can formally describe WSD as the task of assigning the appropriate sense(s) to all or some of the words in T, that is, to identify a mapping S from words to senses, such that $S(i) \subseteq \text{Senses}_D(w_i)$, where $\text{Senses}_D(w_i)$ is the set of senses encoded in a dictionary D for word w_i , and $S(i)$ is that subset of the senses of w_i which are appropriate in the context T. The mapping S can assign more than one sense to each word $w_i \in T$, although typically only the most appropriate sense is selected, that is, $|S(i)| = 1$. So, WSD can be viewed as a classification task: word senses are the classes, and an automatic classification method is used to assign each occurrence of a word to one or more classes based on the evidence from the context and from external knowledge sources. WSD is an AI-complete (Artificial Intelligence complete) problem, which is identical to an NP-complete problem in complexity theory.

II. GUJARATI WORD SENSE DISAMBIGUATION (GWSD) AND APPLICATIONS

A. Gujarati Language

Gujarati is an official and regional language of Gujarat state in India. It is 23rd most widely spoken language in the world today. Gujarati language is spoken by more than 46 million people in India and outside of India that includes Tanzania, Uganda, Pakistan, Kenya and Zambia. Gujarati language belongs to Indo-Aryan language of Indo-European language family and it is also closely related to Indian Hindi language [UCLA]. Gujarati WSD can be formulated as a search problem and solved approximately by exploring the solution search space using heuristic and meta-heuristic algorithms. Several approaches have been investigated for WSD in occidental languages (English, French, German, etc.), including knowledge-based approaches and machine learning-based approaches. However, research on WSD in Gujarati language is relatively limited.

B. Applicattions

There are numerous real-world applications which might get benefit from WSD and can improve their performance.

1. Machine Translation (MT)

The automatic identification of the correct translation of a word in context, that is, machine translation (MT), is a very difficult task. Word sense disambiguation has been considered as the main task to be solved in order to enable machine translation, based on the idea that the disambiguation of texts should help translation systems choose better candidates. In fact, depending on the context, words can have completely different translations. For instance, the English word kite can

be translated in Gujarati as પતંગ, સમડી, કાઇટયલચિત્ર etc. Unfortunately, WSD has been much harder than expected, as we know after years of comparative evaluations.

2. Information Retrieval (IR)

It is very important to resolve ambiguity in a query before retrieving information. As for example, a word depression in a query may have different meanings like illness, weather systems, or economics. And કરમ has different meaning like કૃપા; મહેરબાની, કર્મ; કામ; કૃત્ય, કૃમિ; કરમિયો; પેટમાંપોષાતુંએકજાતનુંજીવડું, કૃમિ, નસીબ; પ્રારબ્ધ; ભાગ્ય; કિસ્મત. So, finding the exact sense of an ambiguous word in a particular question before finding its answer is the most vital issue in this area.

3. Word Processing

Word processing is a relevant application of natural language processing, whose importance has been recognized for a long time. Word sense disambiguation can aid in correcting the spelling of a word, for case change, or to determine when anuswara should be inserted for Indian languages (e.g., in Gujarati, for changing મા (= mother) to માં(= into), or મદ(=proud) to મંદ(=slow), based on semantic evidence in context about the correct spelling).

Given the increasing interest in regional languages in NLP, WSD might play an increasingly relevant role in the determination and correction of words.

III. RELATED WORK

Approaches to WSD belong to two main classes: knowledge based and machine learning based approaches. Knowledge-based methods rely on external lexical resources, such as dictionaries and thesauri, whereas machine learning methods (supervised, unsupervised, or semi-supervised methods) rely on annotated or unannotated corpus evidence and statistical models. Several knowledge-based methods for WSD have been proposed, including gloss-based methods, selectional preferences-based methods, and structural methods. Gloss based-methods consist in calculating the overlap of sense definitions of two or more target words using a dictionary, such as the well-known Lesk algorithm [1]. Other methods use both corpus evidence and semantic relations.

Various metaheuristic algorithms were investigated for the WSD problem. Gelbukh et al. [2] applied a GA to solve it as an optimization problem.

Decadt et al. [3] proposed GAMBL, a word expert approach using a GA to solve WSD. The feature selection and optimization of the parameters of the algorithm are performed jointly using a GA.

Zhang et al. [4] proposed an unsupervised GWSD algorithm and used a GA to maximize the semantic similarity of words.

Menai [5] proposed a GA for the WSD and applied it to an Arabic corpus.

Alsaeedan and Menai [6] proposed a self-adaptive GA for the WSD problem with an automated tuning of its crossover and mutation operators.

Nguyen and Ock [7] introduced a method for WSD using an ant colony optimization algorithm (TSP-ACO).

Manish Sinha, Mahesh Kumar Reddy .R, Pushpak Bhattacharyya , Prabhakar Pandey and Laxmi Kashyap [8], worked on “Hindi Word Sense Disambiguation” that was the first attempt for an Indian language at automatic WSD. The approach is to compare the context of the word in a sentence with the contexts constructed from the WordNet and chooses the winner. The output consisted of a particular Synset number designating the sense of the word. The evaluation was done on the Hindi corpora provided by the Central Institute of Indian Languages.

Neetu Mishra, Shashi Yadav and Tanveer J. Siddiqui [9], “An Unsupervised Approach to Hindi Word Sense Disambiguation” developed an Algorithm that learns a decision list using untagged instances. Some seed instances are provided manually. Stemming has been applied and stop words have been removed from the context. The list is then used for annotating an ambiguous word with its correct sense in a given context. The evaluation has been made on 20 ambiguous words with multiple senses as defined in Hindi WordNet.

Rohan Sharma [10], “Word Sense Disambiguation for Hindi Language” made an attempt to resolve the ambiguity by making the comparisons between the different senses of the word in the sentence with the words present in the Synset form of the WordNet and the information related to these words in the form of parts-of-speech.

Parul Rastogi and Dr. S.K. Dwivedi [11], “Performance comparison of Word Sense Disambiguation (WSD) Algorithm on Hindi Language Supporting Search Engines” compared the performance of WSD Algorithm by using Highest Sense Count (HSC). The Hindi language search engines face the problem of sense ambiguity. The objective is comparative analysis of the WSD algorithm results on the three Hindi language search engines- Google, Raftaar and Guruji.

Neetu Mishra and Tanveer J.Siddiqui [12], “An Investigation to Semi-Supervised approach for Hindi WSD”, investigated Yarrowsky algorithm. After elimination of both, stemming and stop words, the maximum observed precision is 61.7 on 605 test instances.

Sandeep Kumar Vishwakarma and Chanchal Kumar Vishwakarma [13], “A Graph Based approach to Word Sense Disambiguation for Hindi Language” combined Lesk semantic similarity measures and Indegree algorithms for graph centrality and 65.17% accuracy has been obtained.

For proposed approach, the framework is taken as a base from [5].

IV. A PROPOSED APPROACH FOR GWSD USING GENETIC ALGORITHM

A. *Pre-processing*

A text *T* is transformed into a bag of words in a pre-processing stage including mainly the following operations:

1. Tokenization

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens.

2. *Stop-word removal*

Stop word is a word which has less significant meaning than other tokens. For any NLP tools, no single universal list of stop words is used for specific language. Stop words could be important based on its context of application. Some commonly occurring stop words are -છે, તો, પણ, શકાય, હતી, હું, હતા, સાથે,etc.

There exists lot of stopword lists for English and other European languages. There is no standard stopword list for Gujarati language although some efforts are made to prepare stopword list for Guajarati language [14][15].

3. *Stemming*

Stemming is the process of conflating related words to a common stem by chopping off the inflectional and derivational endings. Stemming plays an important role in NLP. This is especially true in case of a morphologically rich language like Gujarati, where a single word may take many forms. The aim is to ensure that related words map to common stem, irrespective of whether or not the stem is a meaningful word in the vocabulary of the language. This can be done by removing prefix, suffix and substituting.

For Example,prefix બિન can be removed from બિનજરુરી and word can be stemmed down to જરુરી. Some examples of prefixes are અનુ, ગેર, સહ, etc. Same way, suffix ેથી can be removed ઘરેથી and word can be stemmed down to ઘર.

Some efforts have been made to develop stemmer for Gujarati language [16][17][18].

B. *GujaratiWordNet*

WordNets have emerged as a very useful resource for computational linguistics and many natural language processing applications. WordNet is a lexical database which comprises of synonym sets, gloss, and position in relations. A synonym set in a WordNet represents some lexical concept. The gloss gives definition of the underlying lexical concept and an example sentence to illustrate the concept.

Since the development of Princeton WordNet [19], WordNets are being built in many other languages. Hindi WordNet was the first WordNet for the Indian language[20]. Based on Hindi WordNet, WordNets for 17 different Indian languages are getting built using the expansion approach. One such effort is GujaratiWordNet. GujaratiWordNet contains Gujarati words used in a family‘s day to day life. It groups words into sets of synonyms called Synsets, provides short definitions and usage examples, and records a number of relations among these Synsets or their members[21][22]. Currently, 26503 nouns, 2805 verbs, 5828 adjectives and 445 adverbs are there in Indo WordNet for Gujarati language.

For Example, consider following sentences. The word **પતંગ** have two Synsets in Gujarati WordNet.

પતંગ નુંલાકડું , છાલઅનેફળોમાંથીલાલરંગનીકળેછે ."
"બાળકોમેદાનમાંપતંગઉડાવીરહ્યાંછે ."

Fig. 1 shows two Synsets with Synset ID, Synonyms, Gloss, POS. WodNet also shows hypernymy, hyponymy and some other relations. Fig. 2 shows the overview of proposed work.

C. *Genetic Algorithm*

Genetic algorithms are inspired by Darwin's theory about evolution. Algorithm is started with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness. This is repeated until some condition (for example number of populations or improvement of the best solution) is satisfied.

Outline of the Basic Genetic Algorithm:

1. [Start] Generate random population of n chromosomes (suitable solutions for the problem)
2. [Fitness] Evaluate the fitness f(x) of each chromosome x in the population
3. [New population] Create a new population by repeating following steps until the new population is complete.
   - I. [Selection] Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
   - II. [Crossover] with a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
   - III. [Mutation] with a mutation probability mutate new offspring at each locus (position in chromosome).

 ISSN: 2321-8169
635 – 639

IV. [Accepting] Place new offspring in a new population

4. [Replace] Use new generated population for a further run of algorithm
5. [Test] If the end condition is satisfied, stop, and return the best solution in current population
6. [Loop] Go to step 2

To formulate the WSD problem in terms of GA, we need to define the following elements:

- A representation of an individual of the population in chromosome.
- Decide a method to generate an initial population and population size.
- Decide an evaluation function to determine the fitness of an individual who can reproduce.
- A description of the genetic operators (crossover and mutation and their rates).
- Methods to select parents for the mating pool and individuals to survive to the next generation.
- Decide termination condition.

The chromosome should in some way contain information about solution which it represents. The most used way of encoding is a binary string.

Chromosome 1	1101100100110110
Chromosome 2	1101111000011110

Each chromosome has one binary string. Each bit in this string can represent some characteristic of the solution. There are many other ways of encoding. This depends mainly on the solved problem. For example, one can encode directly integer or real number; sometimes it is useful to encode some permutations and so on.

To be specific for GWSD, An individual population can be represented by a fixed length integer string. Each gene of an individual is an index to one of possible senses of the word. The initial population can be generated according to one of the following schemes:

4. Random generation: The value of each gene of an individual can be selected randomly from 1 to N using the uniform distribution, where N is the number of possible senses for the corresponding word.
5. Constructive generation: All the senses of a given word are distributed in a round-robin way to the corresponding gene of individuals in the population.

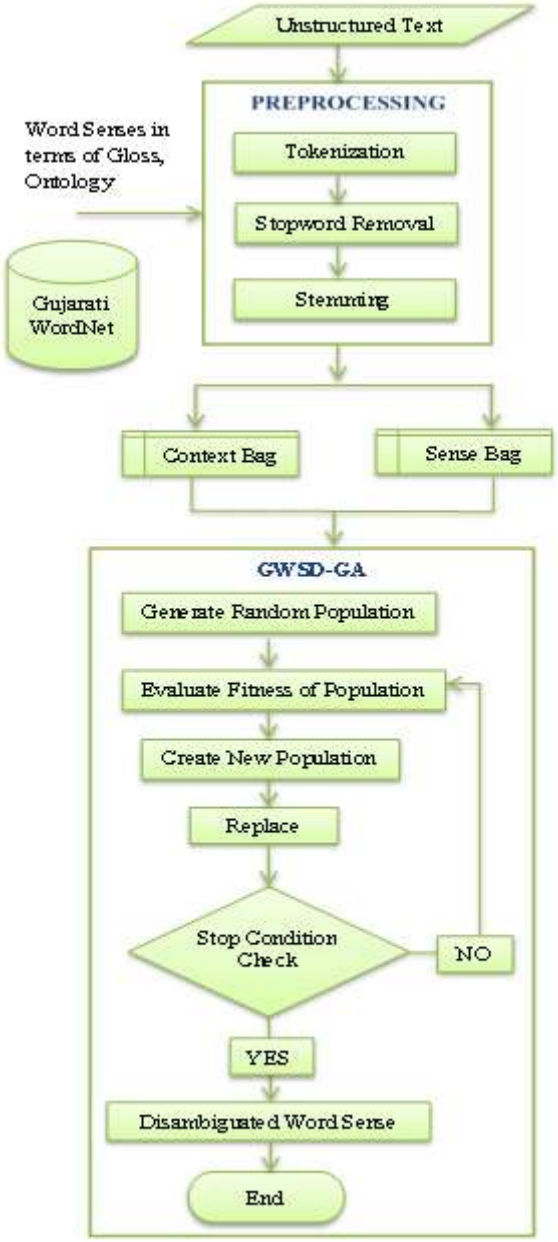

Figure 1. Example of Synsets in Gujarati WordNet

The fitness function can be measured by the word sense relatedness. Crossover selects genes from parent chromosomes and creates a new offspring. Crossover can be rather complicated and depends on encoding of the encoding of chromosome. Specific crossover made for a specific problem can improve performance of the genetic algorithm.Mutation changes randomly the new offspring. The mutation depends on the encoding as well as the crossover.

There are many methods how to select the best chromosomes, for example roulette wheel selection, Boltzman

selection, tournament selection, rank selection, steady state selection, Elitism etc.

V. CONCLUSION AND FUTURE WORK

Word Sense Disambiguation is a very important task in many NLP applications. An extensive research is done inWSD for language like English. There are very few Indian languages like Hindi, Marathi in which the research is done for WSD. There is not much research done for Guajarati Word Sense Disambiguation.

In future work, A Genetic algorithm can be developed and different methods can be investigated to implement the GA components, such as initialization of the population, parent selection, and survivor selection. This prototype can be considered as a framework in which other evolutionary algorithms can be examined as potential approaches to GWSD. Also, it can be made generalised for Indo-Aryan Languages where the Indo-Aryan WordNet for different languages can be used to disambiguate the ambiguous word.

REFERENCES

[1] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone," Proc. 5th annual int. Conf. on Systems Documentation, SIGDOC '86, ACM, New York, NY, USA, 1986, pp. 24-26.

[2] Gelbukh, A., Sidorov, G., Han, S.Y.: Evolutionary approach to natural language word sense disambiguation through global coherence optimization. Trans. Commun. 1(2), 11–19 (2003)

[3] Decadt, B., Hoste, V., Daelemans, W., Bosch, A.V.D.: GAMBL, genetic algorithm optimization of memory-based WSD. In: Proceedings of the Senseval-3: 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, Barcelona, Spain, pp. 108–112 (2004)

[4] Zhang, C., Zhou, Y., Martin, T.: Genetic word sense disambiguation algorithm. In: Proceedings of the 2nd International Symposium on Intelligent Information Technology Application, IITA 2008, vol. 1, pp. 123–127 (2008)

[5] Menai, M.E.B.: Word sense disambiguation using evolutionary algorithms – application to arabic language. Comput. Hum. Behav. 41, 92–103 (2014)

[6] Alsaeedan, W., Menai, M.E.B.: A self-adaptive genetic algorithm for the word sense disambiguation problem. In: Ali, M., Kwon, Y.S., Lee, C.-H., Kim, J., Kim, Y. (eds.) IEA/AIE 2015. LNCS, vol. 9101, pp. 581–590. Springer, Heidelberg (2015)

[7] Nguyen, K.H., Ock, C.Y.:Word sense disambiguation as a traveling salesman problem. Artif. Intell. Rev. 40(4), 405–427 (2013)

[8] Sinha Manish, Reddy Mahesh Kumar, Bhattacharyya R Pushpak, Pandey Prabhakar and Kashyap Laxmi, "Hindi Word Sense Disambiguation", Indian Institute of Technology Bombay, Department of Computer Science and Engineering Mumbai, 2008.

[9] Mishra Neetu, Yadav Shashi and Siddiqui Tanveer J., "An Unsupervised Approach to Hindi Word Sense Disambiguation," Indian Institute of Information Technology, Allahabad. UP, India, 2009.

[10] Sharma Rohan, "Word Sense Disambiguation For Hindi language" Thapar University Patiyala, CSE Dept., India, 2007.

[11] Rastogi Parul and Dr. S.K. Dwivedi, "Performance comparison of Word Sense Disambiguation (WSD) Algorithm on Hindi Language Supporting Search Engines", International Journal of Computer Science Issues, vol. 8, issue.2, March 2011.

[12] Mishra Neetu and Siddiqui Tanveer J., "An Investigation to Semi-Supervised approach for Hindi Word sense Disambigaution", Proceedings of Trends in Innovative Computing 2012- Intellligent System Design, 2012.

[13] Vishwakarma Sandeep Kumar and Vishwakarma Chanchal Kumar, "A Graph Based approach to Word Sense Disambiguation for Hindi Language", International Journal of Scientific Research Engineering & Technology (IJSRET), vol1, issue 5, pp 313-318, Aug. 2012.

[14] Rajnish M. Rakholia, Jatinderkumar R. Saini, "Lexical classes based stop words categorization for Gujarati language", 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Fall)

[15] Hardik Joshi, Jyoti Pareek, Rushikesh Patel, Krunal Chahan, "To stop or not to stop- Experiments on stopword elimination for information retrieval of Gujarati text documents", 2012 Nirma University International Conference on Engineering (NUiCONE)

[16] Nikita Desai, Bijal Dalwadi, "An affix removal stemmer for Gujarati text", 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)

[17] Jikitsha Sheth, Bankim Patel, "Dhiya: A stemmer for morphological level analysis of Gujarati language", 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)

[18] Pratikkumar Patel, Kashyap Popat, Pushpak Bhattacharyya, "Hybrid Stemmer for Gujarati", Proceedings of 1st Workshop on South & Southeast Asian Natural Language Processing (WSSANLP), the 23rd International Conference on Computational Linguistics (COLING), Beijing, August 2010.

[19] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller, ?Introduction to WordNet: An On-line Lexical Database?, CSL Report 43, Princeton University Cognitive Science Laboratory,1990 (Revised August, 1993).

[20] D. Narayan, D. Chakrabarty, P. Pande, P. Bhattacharyya, ?An Experience in Building the Indo-WordNet – A WordNet for Hindi?, In Proceedings of the First International Conference on Global WordNet (GWC 02), Mysore, India, 2002.

[21] Arindam Chatterjee, Salil Rajeev Joshi, Mitesh M. Khapra, Pushpak Bhattacharyya, ?Introduction to Tools for IndoWordNet and Word Sense Disambiguation", Department of Computer Science and Engineering, Indian Institute of Technology Bombay Powai, Mumbai- 400076 Maharashtra, India.

[22] http://www.cse.iitb.ac.in/~pb/papers/gwc12-gujaratiwn.pdf "Introduction to Gujarati WordNet (GCW12) IIT Bombay, Powai, Mumbai-400076 Maharashtra, India.