A DECISION TREE BASED WORD SENSE DISAMBIGUATION SYSTEM IN MANIPURI LANGUAGE

Richard Laishram Singh[1], Krishnendu Ghosh[1], Kishorjit Nongmeikapam[2] and Sivaji Bandyopadhyay[3]

[1]School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar, India
[2]Department of Computer Science and Engineering, Manipur Institute of Technology Manipur, India
[3]Dept of Computer Science and Engineering, Jadavpur University, Kolkata, India

ABSTRACT

This paper manifests a primary attempt on building a word sense disambiguation system in Manipuri language. The paper discusses related attempts made in the Manipuri language followed by the proposed plan. A database, consisting of 650 sentences, is collected in Manipuri language in the course of the study. Conventional positional and context based features are suggested to capture the sense of the words, which have ambiguous and multiple senses. The proposed work is expected to predict the senses of the polysemous words with high accuracy with the help of the suitable knowledge acquisition techniques. The system produces an accuracy of 71.75 %.

KEYWORDS

Word sense disambiguation(WSD); Classification and regression tree (CART); polysemous word; Manipuri.

1. INTRODUCTION

Word sense disambiguation (WSD) is the technique to disambiguate between the senses of a word [1]. It has been recognized as one of the major problems in the field of natural language processing [2]. Word sense disambiguation is not an isolated system by itself, but can be fitted with other important tasks such as, information retrieval, machine translation, speech processing, parts-of-speech tagging and text processing [3, 4]. It has been noted that, the sense of a word is determined based on its syntactic, positional, contextual and other relevant factors. In different scenario, the sense or the meaning of a single word can be different. Such words are known as polysemous words. The current study suggests development of an automatic WSD system [5] with the help of a knowledge acquisition technique.

The goal of the present work is to develop a system which can disambiguate between the different senses of a polysemous word in Manipuri language. In European languages, efficient and automatic WSD systems are present. In Indian languages, such systems are mostly rule-based due to lack of standardized database and presence of the proper knowledge acquisition tools. So far, negligible amount of work has been reported in Manipuri language. That motivates the present attempt to collect a suitable database consisting of every possible contexts and senses, used in day-to-day life.

DOI:10.5121/acij.2014.5403

The current study reports mainly the development issues of a WSD system in Manipuri language. Manipuri is basically a Tibeto-Burman language, spoken mainly in the valley of Manipur, a North-Eastern state of India. Hence, the syntactic and semantic structures of the language are different from other Indian languages. The task of disambiguating the senses in Manipuri language poses a challenge due to its agglutinative, reduplicative, compounding and tonal nature [6]. The language uses the traditional Meitei-Mayek script for general use. But, that script is not used in the current study due of presence of limited words and structures. The present work uses Bengali script for developing the database. The lack of a standard font made the task even more challenging. Besides, limited amount of works are reported in Manipuri language in the field of natural language processing such as, POS tagging [7], Transliteration [8] and negligible amount in WSD.

The paper is organized as follows: related literatures are discussed in section 2. The details of the corpus are mentioned in section 3 The proposed architecture of the word sense disambiguation module is discussed in the section 4 followed by the conclusion and direction towards future work in section 5. Finally a few important references are added.

2. LITERATURE SURVEY

Several works on WSD are reported in English and other European languages. On the contrary, limited works have been noted for Indian languages. The approaches are classified in five groups: (i) selection restriction based disambiguation, (ii) machine learning based approach, (iii) supervised learning approach, (iv) bootstrapping approach and (v) dictionary based disambiguation.

2.1 Selection restriction based disambiguation

Selection restriction and type hierarchies are the primary knowledge sources used to disambiguate the senses of the words in most of the integrated approaches. In an integrated rule based approach for semantic analysis [9], selection restrictions block the formation of component meaning representations containing the selection restriction violations [1]. For Indian languages, where the semantic and syntactic structures are not analyzed properly and the required knowledge sources are missing, rule based processes generally are not suggested.

2.2 Machine learning based approach

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

2.3 Supervised learning approach

Supervised learning is a specific category of the machine learning approaches to train the classifier from supervised (labelled) training data [1]. The classifier is fed with the training data consisting of the exemplars along with the output labels. The classifier predicts the correct output value for any valid test input object.

2.4 Bootstrapping approach

Bootstrapping approach is an incrementive machine learning based approach, widely used for developing word sense disambiguation systems. Bootstrapping approach eliminates need for a large training set by relying on a relatively less number of instances called seeds. Seeds are used for training and testing. The generated pair of test patterns and the predicted output is later added with the seeds to populate the training data. This process is very useful, if the base database is comparatively small in size.

2.5 Bootstrapping approach

Dictionary based disambiguation approaches use a dictionary which provides both the means of constructing a sense tagger and the target sense [1, 10]. In order to perform a large scale disambiguation, machine readable dictionaries (MRD) are used to automate the task. This process is useful for the languages having a digitalized machine readable dictionary. Due to the lack of standard database and related knowledge acquisition tools, such approaches can hardly be used for the focus Manipuri or other Indian languages.

3. DATABASE COLLECTION

The database used in the current study contains a total of 672 Manipuri sentences. These sentences are collected from a local newspaper "The Sangai Express". The sentences contain a total of 13,167 words and around 2,000 polysemous words. Polysemous words are the focus of the current study as they contain more than one sense based on contexts and other factors. The sentences are chosen from the news domains to accommodate day-to-day words and senses mostly. The entire database, due to lack of digital resources and standard font are typed manually [11]. The database is thoroughly checked for inconsistencies present in spelling, spacing and, punctuation. The polysemous words are then tagged with the sense manually.

4. PROPOSED ARCHITECTURE

The current study attempts to find a solution to disambiguate between the possible senses of Manipuri words. A base database is collected with sentences consisting of day-to-day senses. The database is then preprocessed and suggested features are collected out of it. Then, a classifier is developed with the knowledge obtained from the features using a learning algorithm. The classifier finally predicts the sense of the test sentences.

The word sense disambiguation system works in two phases: (i) training phase and (ii) testing phase. In the training phase, the training data is first preprocessed and features are generated. The features are used then to train the classifier based on any learning algorithm. On the contrary, in the testing phase, the test data is preprocessed and features are generated. The features of the test data are fed to the classifier and the predicted output is compared for performance evaluation of the system. Hence, the proposed architecture of the word sense disambiguation module for Manipuri language contains five building blocks: (i) preprocessing, (ii) feature selection and generation and (iii) training, (iv) testing and (v) performance evaluation.

*A. **Preprocessing***

Preprocessing is the module where the raw data is processed so that the features can be generated from the training or test data efficiently. Data is converted to corresponding ASCII format after

the inconsistencies (like spelling, punctuations and spacing mistakes) are removed. When the data is converted to ASCII format, the selected features are generated semi-automatically.

B. Feature Selection and Generation

In this phase, the ASCII format generated database is used to build the feature. The current study, as it is in the preliminary attempt, uses a set of very common, widely popular and easily extractable features. A total of 6 features are taken to build feature:

(i) the focus word for which the sense is to be derived,
(ii) the normalized position of the word in the sentence,
(iii) the previous word,
(iv) the previous-to-previous word,
(v) the next word,
(vi) the next to next word.

A 5-gram window is formed using the pair of the focus word and its context words which forms the context information. A focus word, based on the context may have different senses. Hence, in order to disambiguate the sense of the focused word, the contextual information is very much necessary [13], and helps in predicting the correct one. In the current study positional feature is suggested because of the lack of other relevant morphological features. As the syntactic and semantic structures of a sentence remain mostly similar for a particular language, this feature contains probable morphological information. To generate the final input feature vector, from the database mentioned above mentioned six features are collected automatically.

C. Training

In the present work, a supervised learning algorithm is proposed based on decision tree algorithm. In supervised approaches, the classifier is developed with the help of the input feature vector and the output labels of the data. Hence, the final feature vector is developed using the six features mentioned in section B and the output sense of the focus word. The sense of the focus word is derived manually and finally the seven entries are fed to the classifier. The classifier is trained with the decision tree algorithm.

The decision tree is a simple and linear decision making approach. In the current work, classification and regression tree (CART) based algorithm is suggested to train the classifier. Binary decision tree is developed automatically using the seven entries of the focus word present in training data. CART uses all the instances of the training data and asks binary questions on the features. CART algorithm selects the most predictive feature from the feature set and the best possible question to achieve accurate classification of the training. Based on the feature and its distribution of values, the training data is segmented into two parts. This segmentation is carried out based on each of the features and finally at the leaves, a output class is achieved. CART has been reported as one of the suitable learning algorithms for developing supervised learning models for Indian languages like Manipuri whose characteristics have not studied in detail. An example of a CART is given in Fig 1.

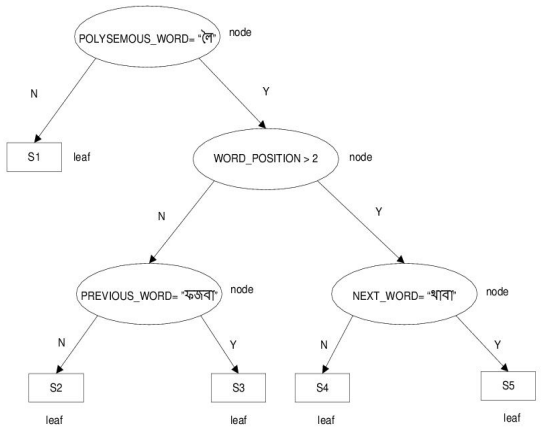

Fig 1: Flow Chart of CART based word sense disambiguation

D. Testing

During the testing, CART offers prediction by comparing the features for the test case with the trained decision tree. While predicting the sense for a test word, the corresponding features are generated and compared with the trained decision tree. Starting from the root node asking a question on individual features, the features of the test word will lead to a leaf which offers an approximated output.

The features are already generated in the feature generation phase. The trained CART is fed with features are test data and corresponding approximations are noted. The predictions are later compared with the correct sense tags to perform evaluation of the current system.

E. Performance Evaluation

The accuracy of the proposed word sense disambiguation system is evaluated using an objective analysis. The analysis is carried out by determining the correct sense prediction percentage for the test words.

Being trained on 1,600 words and tested on 400 words using 7 features, the word sense disambiguation system predicts the senses with 71.75% accuracy. The performance of the proposed model is discussed in TABLE 1.

Predicted Sense	Correctly Predicted Sense	Accuracy
400	287	71.75%

TABLE I: Objective Evaluation of CART based word sense disambiguation model

5. CONCLUSION AND FUTURE WORK

This paper proposed a preliminary attempt to predict the senses of the words for Manipuri language. A set of positional and contextual features are suggested for developing the word sense disambiguation system. Due to unavailability of morphological and other relevant linguistic

knowledge, the accuracy of the model is lower than the models built for languages like English. Moreover, the size of training data or the features used are not sufficient to achieve the classification quality of human transcribers. A relevant and appropriate feature set can achieve higher accuracy. Further improvements are expected by:

1. Joining a rule-based model with the CART based data-driven model to improve accuracy of the word sense disambiguation system.
2. Exploring other well-known classifiers such as ANN, SVM or other probabilistic methods like HMM and N-gram models.

ACKNOWLEDGEMENTS

The work presented in this paper is performed at KIIT University as a project for the fullfilment of the degree of Masters of Technology. Special thanks to Mr. Kishorjit N. Singh for his help in data collection.

REFERENCES

[1] D.Jurafsky, J.H.Martin, "Speech and Language Processing,", Pearson Publishers, December, pp. 658-672
[2] R.Mihalcea, D. Moldovan, "Word Sense Disambiguation based on Semantic Density", in proceedings of Seventh International conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2010, pp. 1492-1496.
[3] M.Sinha, M.K.Reddy, P.Bhattacharyya, P.Pandey and L.Kashyap, "Word Sense Disambiguation," in proceedings of International Journal of Computer Applications (IJCA),2010, vol. 5.9, pp. 25-32.
[4] R.Mihalcea and D. Moldovan, "A Method for Disambiguating Word Senses in a Large Corpus," in proceedings of COLING/ACL Workshop on Usage of WordNet in Natural Language Processing, Computers and the Humanities, 1992, volume 26, no. 5-6,pp. 415-439.
[5] C.Leacock, G.Towell and E.Voorhees, "Corpus-Based Statistical Sense Resolution," ARPA Workshop on Human Language Technology, 1993.
[6] T.D.Singh and S.Bandyopadhyay, "Word Word Class and Sentence Type Identification in Manipuri Morphological Analyzer," in proceedings of MSPIL, 2006.
[7]K.Nongmeikapam, L.Nonglenjaoba, Y.Nirmal and S.Bandhyopadhyay, "Improvement of CRF based Manipuri POS tagger by using Reduplicated MWE (RMWE)," arXiv preprint arXiv: 1111.2399, 2011.
[8] K.Nongmeikapam and S. Bandhyopadhyay, "A Transliteration of CRF Based Manipuri POS Tagging," in proceedings of Procedia Technology, Elsevier, 2012, pp. 582-589.
[9] M S. Olsen, "WordNet Word sense Disambiguation using an Automatically Generated Ontology," Class of 2003 Senior Conference on Natural Language Processing, 2003.
[10] J.Veronis and M. Nancy, "Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionarie," in Proceedings of 13th conference on Computational linguistics, 1990, volume 2, pp. 389-394.
[11] M.M. Khapra, S. Shah, P. Kedia and P. Bhattacharyya, "Projecting Parameters for Multilingual Word Sense Disambiguationin Proceedings of 2009 Conference on Empirical Methods in Natural Language Processing, 2009,volume 1, pp. 459-467.
[12] K.Ghosh, R. V. Reddy, N. P. Narendra, S. Maity, S. Koolagudi and K. S. Rao, "Grapheme-to-phoneme Conversion in Bengali for Festival based TTS Framework", in Proceedings of International Conference of Natural Language Processing, 2010, pp. 294.
[13] A.Eneko and G.Rigau, "Word Sense Disambiguation using Conceptual Density", in Proceedings of the 16th International Conference on Computational Linguistics (COLING), Copenhagen, Denmark, 1996.