A Supervised Approach on Gurmukhi Word Sense Disambiguation using k-NN Method

Himdweep Walia
Assistant Professor
GNIOT, Gr. NOIDA
Uttar Pradesh, India
himdweep@yahoo.com

Ajay Rana
Professor
Amity University, UP
Uttar Pradesh, India
ajay_rana@amity.edu

Vineet Kansal
Professor
CSED, IET, Lucknow
Uttar Pradesh, India
vineetkansal@yahoo.com

***Abstract* – The primary objective of Natural Language Processing (NLP) is to ensure that communication is established between the human and the machine. An ambiguous word is the one which has more than one meaning. The purpose of Word Sense Disambiguation (WSD), an important area of NLP, is to ensure that the machine is able to correctly find the context in which the word is being used. A number of different supervised, semi-supervised and un-supervised algorithms are being employed to carr y out the same. One such supervised approach is k–NN algorithm which we have implemented in disambiguating words in Gurmukhi. Gurmukhi (or popularly known as Punjabi) is the 17th most spoken language in the world. For this paper we have used the Punjabi Corpora (obtained from Evaluations and Language Resources Distribution Agency, Paris, France) which has been sense-tagged with 100 words.**

Keywords – Word Sense Disambiguation, Supervised Approach, k-Nearest Neighbor, WSD in Punjabi, Sense-tagged corpora

I. INTRODUCTION

The Natural Language Processing (NLP) is one of the key research areas in Artificial Intelligence. Topics like Natural Language Understanding and Natural Language Generation comes under this. The study of Word Sense Disambiguation (WSD) describes how the machine can easily understand the context in which the given word is being used. An ambiguous word has more than one meaning, which a human can easily understand when using the given word. For example, the word, "left" tells the direction i.e. opposite of right and is also the past tense of the word, leave. Given the context, the human can righty distinguish between the two. The same intelligence is to be provided to the machine to understand the difference.

All the natural languages spoken in the world have a huge number of ambiguous words. Like for example, in our language of study, i.e. Gurmukhi (popularly referred as Punjabi), we have multiple ambiguous words. The word,

ਹਾਰ

has five different meanings. Consider the following instances:

Instance 1: ਮੈਂ ਫੁੱਲਾਂ ਦਾ ਹਾਰ ਖਰੀਦਿਆ |

Instance 2: ਉਸ ਨੂੰ ਤਾਂ ਵਪਾਰ ਵਿਚ ਬਹੁਤ ਹਾਰ ਹੋਇ |

Instance 3: ਰੱਬ ਕੁਦਰਤ ਦਾ ਸਿਰਜਣਹਾਰ ਹੈ |

Instance 4: ਮੈਂ ਤਾਂ ਮੇਹਨਤ ਕਰ ਕੇ ਹਾਰ ਗਿਆ ਹਾਂ |

Instance 5: ਅਸੀਂ ਮੈਚ ਹਾਰ ਗਏ |

In the first instance the word "haar" means "garland", in the second instance it means "unsuccessful", in the third instance it means "doer", in the fourth instance it means "give up" and in the fifth instance it means "loose".

There are number of algorithms that are being implemented under WSD. These algorithms have been broadly classified into three main categories: knowledge-based algorithms, machine-learning based algorithms and hybrid algorithms. The knowledge-based algorithms are based on machine-readable dictionaries also referred to as thesauri. The idea is to list out different meanings of the words and by using one of the knowledge-based algorithms, the machine is able to deduce the correct context of the word. The second category, i.e. machine learning based algorithms are further sub-categorized into three, which is, supervised algorithms, unsupervised algorithms and semi-supervised algorithms. The supervised algorithms are based on sense-tagged corpora, better known as the training sets, where with the help of a classifier the correct meaning of the given word is sought. In the unsupervised techniques, there is no need for a sense-tagged corpora. These set of algorithms try to deduce the correct meaning of the word by accounting the clusters of words around the given word. The semi-supervised algorithms follow the middle path where a small sense-tagged data is used as the reference and using this the classifier is trained to produce similar

978-1-5386-1719-9/18/$31.00 ©2018 IEEE

sense-tagged corpora and subsequently the whole text is tagged. The third major category which is the set of hybrid algorithms is the amalgamation of all the above algorithms. Every algorithm has its own sets of merits and demerits. The experiments that have been conducted using supervised algorithms have yielded better results, provided a sense-tagged corpora is available.

The k- Nearest Neighbor (k-NN) Algorithm is one of the algorithms under the category of Supervised Algorithm. This algorithm has been explored to disambiguate Punjabi words, for this paper. Punjabi is one of the regional languages spoken and written in India. The work done on the regional languages has till now been limited to preparation of machine-readable dictionaries and translation process. The major reason for this is the unavailability of sense-tagged corpora for Punjabi language. In this paper, for training and testing, we are using the Punjabi Corpora (obtained from Evaluations and Language Resources Distribution Agency, Paris, France) which has been sense-tagged with 100 words.

The rest of the paper has been divided in the following manner: Section II discusses the related work in other languages. Section III explains the proposed supervised algorithm i.e. k-NN. Section IV presents the observations and its interpretation and the last section i.e. Section V summarizes the paper.

II. PREVIOUS WORK

A lot of extensive work has been done in English and other European languages and even on Asian languages like Japanese and Chinese in the domain of Word Sense Disambiguation (WSD) [7]. In recent years, a lot of work has been carried out in Indian Regional Languages as well [1]. Our language of focus is Gurmukhi, more popularly known as Punjabi [7]. The major amount of work in this language has been limited to machine translation and building of machine readable dictionaries due to lack of sense-tagged corpora [21]. For this paper, we have studied the work done in Hindi Language as Gurmukhi drives closely to this language [12, 13, 14, 15, 1 6]. The k-NN algorithm has been implemented in English Language [24], which has been read for this paper. Among the Indian Regional Languages, this algorithm has been used for Bengali Language [23] and that paper has been used as base paper for this study.

In their paper, A. R. Rezapour, S. M. Fakhrahmad, M. H. Sadreddini [24], have extracted two set of features – one, the set of words that have frequently occurred in the given text, and second, the cluster of words surrounding the given ambiguous word. Then they have applied 5 fold cross validation process to divide the data into training and testing part for the k-NN classifier. In this paper they have also proposed a feature weighing strategy which has shown commendable results in most of the experiments undertaken than the standard procedure.

In their paper, R. Pandit, S. K. Naskar [23], have used the supervised technique of k-NN algorithm for disambiguation in Bengali. In their paper they have calculated the distance between the two vectors by using the overlap metric, which helped them to get an accuracy of 71%. Their experiment was conducted on 100 testsets sentences where 25 target words, which included 10 nouns, 4 adjectives, 8 verbs and 3 adverbs, were disambiguated.

III. k-NN BASED WSD APPROACH

We are using one of the algorithms under the supervised approach - k-NN algorithm - for WSD. In this algorithm we classify the given test data which is similar in meaning from the training data. This is based on the learning by analogy [24]. When an unprocessed data, which we refer as vector, so when an unknown vector is given, the k-NN classifier finds the similar vector from the training set. These training vectors are the "nearest neighbors" of this unknown vector.

The "k" in the k-NN, is a positive integer number whose value can be calculated experimentally. If the value of k is 1 then the unknown vector is assigned to the class of its nearest neighbor, otherwise it is classified by the majority vote of its neighbors.

We have to mathematically calculate the distance between the set of two nearest neighbors i.e. the unknown vector and the training vector. This distance determines the closeness of the unknown vector with the given set. The distance between the two vectors can be calculated from any of the following methods: Hamming Distance, Euclidean Distance, Manhattan Distance, Overlap Metric, etc.

In this paper, we are using the Euclidean Method to calculate the distance. The Euclidean Distance between two vectors, say $X_1 = (x_{11}, x_{12}, \ldots, x_{1n})$ and $X_2 = (x_{21}, x_{22}, \ldots, x_{2n})$, is calculated as

$$dist\,(x_1, x_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \quad (1)$$

The proposed methodology includes two steps: the first step performs the feature extraction process and converts the given paragraph of the corpus into a vector. In the second step we pass these vectors through the k-NN classifier.

A. Feature Extraction

The first step that we undertook for developing the WSD system was feature extraction is the first step in the development of WSD system. In this stage, first of

all we remove all the stop-words i.e. prepositions, pronouns, conjunctions, etc. from the context. After covering this pre-processing stage, the given context is analyzed to get two sets – one, set of frequent words, and second, set of surrounding words.

Set of frequently used words

In this, we extract the most frequently appearing words in the context. This set of words, referred as feature, forms the first set. We calculate the value of this feature based on the number of times these set of words have been used with the given ambiguous word for the given context.

Set of words surrounding ambiguous word

In this process, we select '*n*' words on either side of the ambiguous word. The value of *n* is determined experimentally. The set of words which co-occur more frequently with the given ambiguous word are selected for the dataset. The value of this feature is calculated by counting the number of words between the two i.e. our ambiguous word and the surrounding word.

B. Using k-NN Classifier

After we have extracted the features, we construct the dataset using these extracted features i.e. set of frequent words and set of surrounding words along with the set of ambiguous words and their various senses.

The next step is to select a sentence having the ambiguous word (with k^{th} sense) from the given corpora as test instance. Then the k-NN algorithm is applied on these two sets where the test vector is compared with all the training vectors to find the sense closest to the real meaning of the ambiguous word.

ALGORITHM

k-NN approach was applied to the corpora using the following steps:

Given:
(a) Set of ambiguous words with their sense
(b) Set of frequently used words (with every ambiguous word)
(c) Set of surrounding words (for every ambiguous word)
(d) Punjabi Corpora (obtained from Evaluations and Language Resources Distribution Agency, Paris, France) which has been sense-tagged with 100 words.

Step 1: Start
Step 2: A sentence having an ambiguous word is selected as an input.
Step 3: Remove the stop words from the given input.
Step 4: Testset is constructed from the input sentence.
Step 5: For a given ambiguous word in its k^{th} sense
DO
Calculate the distance of the testset w.r.t. set of frequent words, using Euclidean Distance, call it f_i
AND / OR
Calculate the distance of the testset w.r.t. set of surrounding words, using Euclidean Distance, call it s_i
Step 6: Form two lists:
List a – Sorted in descending order of the distances between the testset and set of frequent words.
List b – Sorted in descending order of the distances between the testset and set of surrounding words.
Step 7: The value of *k* is to be selected such that k > 0.
Step 8: Select the 'k' nearest neighbor
Step 9: Select from the given list (either 'a' or 'b') the training vector which is nearest to the given test vector.
Step10: Stop

IV. OBSERVATION AND INTERPRETATION

For our paper, we experimented the efficiency of the given algorithm on 120 testset sentences with 8 ambiguous words. The results of the k-NN based approach to WSD in Gurmukhi is as given in Table 1. The table shows the results with respect to two different testsets, one consisting of frequent words (List a) and the other consisting of frequent words (List b).

We used the 5 fold cross-validation [24] to estimate the performance of the algorithm. Thus, for every ambiguous word, the set of all related samples (List a and List b) were divided into 5 equal folds. The idea was to use any four folds to extract the features so that we are able to train the k-NN classifier and use the fifth fold for testing purpose. We repeated this process five times ensuring that every fold is used as the test data at least once. The average accuracy has been listed in given table.

TABLE 1: ACCURACY VALUES OBTAINED ON LIST a AND LIST b USING THE k-NN APPROACH

Ambiguous Word	List a	List b
ਉੱਤਰ	76.2	76.4
ਸੰਗ	72.4	72.4
ਸਹੀ	68.8	69.8
ਹਾਰ	71.2	72.4
ਕੱਚਾ	74.4	76.2
ਘੱਟ	53.8	53.6
ਚੱਕ	72.6	72.6
ਜੋੜ	74.4	76.2

V. CONCLUSION

In our paper, we have proposed a supervised learning algorithm for Word Sense Disambiguation based on k-NN approach. We extracted two set of features; the set of words that have occurred frequently along with the ambiguous word in the corpora and the set of words surrounding the ambiguous word in the corpora. Then, using the 5 fold cross-validation process, we have divided the given data into two i.e. training set and test set for the k-NN classifier where the experiment was separately carried out with respect to the two set of features.

ACKNOWLEDGEMENTS

The author[1] is grateful to the Management of Greater Noida Institute of Technology, Gr. Noida, which graciously gave permission to pursue doctorate and have also provided necessary support in that regard.

The author[1] also acknowledges Amity University, Uttar Pradesh, where the author is registered to pursue doctorate and has been provided with the library facility and extensive online resources.

REFERENCES

[1] M. Bansal, "Word Sense Disambiguation: Literature Survey for Indian Languages", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 12, December 2015.

[2] J. Kaur, "Word Sense Disambiguation (WSD)", International Journal For Technological Research In Engineering, Volume 1, Issue 5, January 2014.

[3] M. M. Khapra, S. Shah, P. Kedia, and P. Bhattacharyya, "Projecting Parameters for Multilingual Word Sense Disambiguation", Proceedings of the Conference on Empirical Methods in Natural Language Processing, Singapore, 2009.

[4] R. Kumar, and R. Khanna, "Natural Language Engineering: The Study of Word Sense Disambiguation in Punjabi", IJES, July 2011.

[5] R. Kumar, R. Khanna, and V. Goyal, "A Review of Literature on Word Sense Disambiguation", International Journal of Engineering Sciences, Vol. 6, July 2012.

[6] A. Narang, R. K. Sharma, and P. Kumar, "Development of Punjabi WordNet", Springer, CSIT, December 2013.

[7] R. Navigli, "Word Sense Disambiguation: A Survey", ACM Computing Surveys, Vol. 41, no.2, Article 10, February, 2009.

[8] P. Rana, and P. Kumar, "Word Sense Disambiguation for Punjabi Language using Overlap Based Approach", Chapter in Advances in Intelligent Informatics, Volume 320 of the series Advances in Intelligent Systems and Computing, pp 607-619, Springer.

[9] J. Singh, and I. Singh, "Word Sense Disambiguation: Enhanced Lesk Approach in Punjabi Language", International Journal of Computer Applications, Volume 129 – No.6, November 2015.

[10] S. Singh, and T. J. Siddiqui, "Role of Karaka relations in Hindi Word Sense Disambiguation", Journal of Information Technology Research (JITR), IGI Global, Volume 8, Issue 3, 2015, pp. 21 - 42.

[11] S. Singh, and T. J. Siddiqui, "Role of Semantic Relations in Hindi Word Sense Disambiguation", Proceedings of the International Conference on Information and Communication Technologies (ICICT 2014), Kochi, India, 3-5 December, 2014, Elsevier Procedia Computer Science, Volume 46, 2015, pp. 240-248.

[12] S. Singh, V. K. Singh, and T. J. Siddiqui, "Hindi Word Sense Disambiguation using Semantic Relatedness measure", Proceedings of 7th Multi-Disciplinary Workshop on Artificial Intelligence (MIWAI 2013), 9-11 Dec. 2013, Krabi, Thailand, pp. 247-256, LNCS, Springer.

[13] S. Singh, and T. J. Siddiqui, "Utilizing Corpus Statistics for Hindi Word Sense Disambiguation", International Arab Journal of Information Technology (IAJIT), SCI Expanded, Volume 12, No. 6A, December 2015, pp. 755 – 763.

[14] S. Singh, and T. J. Siddiqui, "Evaluating Effect of Context Window Size, Stemming and Stop Word Removal on Hindi Word Sense Disambiguation" , Proceedings of the International Conference on Information Retrieval & Knowledge Management, CAMP2012, 13-15 March, 2012, Malaysia, pp.1-5, IEEE Explorer.

[15] H. Walia, A. Rana, and V. Kansal, "Different Techniques Implemented in Gurumukhi Word Sense Disambiguation", International Journal of Advanced Technology in Engineering and Science, Volume 05, Issue 06, June 2017.

[16] J. Kaur, and V. Gupta, "Effective Approaches for extraction of Keywords", International Journal of Computer Science, Issue 6, Vol. 7, November 2010.

[17] R. Pandit, and S. K. Naskar, "A Memory Based Approach to Word Sense Disambiguation in Bengali Using k-NN Method", 2nd IEEE International Conference on Recent Trends in Information Systems, 2015.

[18] A. R. Rezapour, S. M. Fakhrahmad, and M. H. Sadreddini, "Applying Weighted KNN to Word Sense Disambiguation", Proceedings of the World Congress on Engineering, Vol III, WCE 2011, July 6 - 8, 2011.